Feedback Loops in Interactive Machine Learning: Online Weakly-Submodular Learning and Probing for Missing Labels

Adhyyan Narang

A dissertation

submitted in partial fulfillment of the

requirements for the degree of

Doctor of Philosophy

University of Washington

2026

Reading Committee:

Lillian J. Ratliff, Co-Chair

Maryam Fazel, Co-Chair

Kevin Jamieson

Program Authorized to Offer Degree:

Electrical and Computer Engineering

University of Washington

Abstract

Feedback Loops in Interactive Machine Learning:
Online Weakly-Submodular Learning and Probing for Missing Labels

Adhyyan Narang

Co-Chairs of the Supervisory Committee:
Lillian J. Ratliff
Department of Electrical and Computer Engineering
Maryam Fazel
Department of Electrical and Computer Engineering

Machine learning systems are increasingly deployed as interactive services that obtain data not by sampling from a fixed distribution, but through direct and indirect interaction with an environment, users, and other learners. In recommender engines, language model services, and online platforms, this interaction makes the learner's information environment *endogenous*: the learner's own actions or current state determine what feedback and data become available to it. This dissertation studies two distinct channels through which interactivity induces endogeneity, and develops principled algorithms with provable guarantees for each.

Chapter I addresses *history-dependent feedback in repeated interaction.* When a learner constructs a set of choices over time (for instance, recommending movies sequentially), the value of each future action depends on what has already been selected: a sequel gains value if the original was recommended, while similar items exhibit diminishing returns. The learner's past actions shape the structure of its own future feedback, creating combinatorial utilities that are neither purely submodular nor purely supermodular. We extend Gaussian Process contextual bandits to objectives that are *BP-decomposable* (a sum of monotone submodular and supermodular terms) or *weakly submodular*. We introduce a novel separate-feedback framework where observations are available

independently for each component, and integrate Nyström sketching to ensure scalability. We prove sublinear regret bounds in all cases, demonstrating that richer utility structures can be optimized online with theoretical guarantees.

Chapter II addresses *choice-driven data allocation in multi-learner markets*. When multiple learners compete for the same pool of users, who choose based on predictive quality and inherent preferences (e.g., brand loyalty), the data each learner observes becomes a function of its own performance, creating a second form of endogeneity. We characterize an *overspecialization trap*: as learners optimize for users who already prefer them, they become less attractive to others, further restricting their data and leading to arbitrarily poor global performance, even when models with low full-population loss exist. Inspired by knowledge distillation, we propose *Peer Probing*, an algorithm that queries peer models to obtain synthetic labels for users outside the learner's organic base. We prove that this procedure converges almost surely to a stationary point with bounded full-population risk when probing sources are sufficiently informative.

Together, these contributions show that accounting for the endogeneity inherent in interactive learning, through richer function classes and richer data sources, yields algorithms that are both theoretically principled and practically effective.

Contents

List of Figures

List of Tables

ACKNOWLEDGEMENTS

First and foremost, I would like to express my gratitude to my advisors, Lillian Ratliff and Maryam Fazel, for their support throughout my doctoral journey. If it were not for Lily, I would never have been introduced to game theory, which has now become a core component of my academic worldview and identity. Through working together both at UW and Amazon, I learned a great deal from Lily, from best practices in writing proofs to developing a taste for good theoretical problems that are practically motivated. Maryam's experience and research judgment have been extremely beneficial for me across all of my projects. From her, I have learned what a good theoretical problem looks like, the importance of precise formulations, and the abstract but invaluable skill of prioritizing promising directions during research.

During my PhD, I also had the good fortune to receive guidance from a number of other mentors. I would like to thank Kevin Jamieson for providing valuable support throughout my PhD. Through his class on interactive learning and our collaboration on instance-dependent RL, I broadened my horizons past supervised learning theory and moved towards the settings that now comprise a bulk of this thesis. I would like to thank Dima Drusvyatskiy: almost my entire statistical toolbox comes from the classes I took with Dima. Seeing these tools in action in the performative prediction problem was a very satisfying and didactic experience. I would like to thank Jeff Bilmes for introducing me to the underappreciated and ubiquitous class of submodular functions, for which Jeff has a contagious passion. I would like to thank Sarah Dean; during our meetings, Sarah always had a way of getting to the heart of the theoretical question I was tackling irrespective of how incoherent my description of my obstacle may have been. I am grateful as well to Samet Oymak, who has a way of making theoretical research very enjoyable, even in the moments when we were stuck. I would also like to thank Vidya Muthukumar and Anant Sahai. Their mentorship made it possible for me to become a theoretical researcher, and in many ways helped me begin the journey that culminated in this

dissertation. Through the guidance of this wonderful set of mentors and collaborators, I was able to significantly expand my theoretical capacity by taking on increasingly challenging problems, and refine my conceptual and writing skills.

Next, I would like to thank Andrew Wagenmaker and Omid Sadeghi, who provided me with valuable mentorship during my projects. I am extremely grateful to everyone in the WAIL Pods Group, my core intellectual community throughout my PhD; it always served as a space where I felt both comfortable and intellectually challenged. In particular, I would like to thank my collaborators, labmates and friends: Evan Faulkner, Tanner Fiez, Ben Chasnov, Jason Isa, Yue Sun, Dan Tabas, Avi Bose, Arnab Maiti, Claire Zhang, Artin Tajdini.

I would like to thank Amazon for generously supporting me through the Amazon Hub Fellowship and two enriching internships. In particular, I would like to thank my mentors: Arbaaz Khan, Anurag Beniwal, Amanda Bouman and Josh Hooks. Through these internships, I developed a broader perspective for where my theory work fits in machine learning practice and what types of theoretical problems are valuable. Moreover, I developed a complementary skillset of tackling applied problems which made me a much more well-rounded researcher and better prepared for my next stage as a researcher.

I would like to thank the Seattle Insight Meditation Society and Halea Yoga, which provided me with refuge during challenging times and confidence during positive times to stretch my limits. I would like to thank all my friends with whom I had the good fortune to share many positive memories and explore everything that Seattle has to offer. In particular, I am grateful to Saman Salike, Arijit Nerurkar, Neeha Kotte, Dhruv Singh, Sunny Bakhda, and Nidhi Jaltare and my wonderful set of housemates in 4304 for helping me feel at home in Seattle, despite being so far away from my home and family in Mumbai.

Lastly, I would like to thank my family. My parents always encouraged me to carve my own path, and always supported me through the ups and downs of doing so. None of this would have been possible if it were not for their love and support.

Chapter 1

Introduction

A decade ago, the dominant paradigm in machine learning was largely static: collect a fixed dataset, train a model, and evaluate it on a held-out test set. The foundational theory that accompanied this paradigm consisted of uniform convergence generalization bounds, estimation rates, and convex optimization guarantees; these treated the data source as exogenous, a background assumption rather than an object of study. This abstraction enabled deep progress, but it also reflected the reality of how most models were built and deployed at the time.

That reality has changed substantially. Machine learning systems are now deployed as ongoing, interactive services that obtain data not by sampling from a fixed distribution, but through direct and indirect interaction with an environment, users, and other learners. Recommendation and ranking engines are routine infrastructure in media, commerce, and online platforms [71, 89]. Language model services have become everyday interfaces, with multiple providers competing for the same users [80]. These systems do not fit a model once; they shape user behavior and continuously receive feedback from it through engagement signals, ratings, prompts, and other behavioral traces. Reinforcement from human feedback has made user interaction a direct training signal, and knowledge distillation and synthetic data training [24, 49, 101, 107] mean that models increasingly learn from each other's outputs, not just from organic user data. Feedback loops between a learner and its data source, endogenous data distributions, and multi-agent training dynamics are no longer edge cases; they are basic features of how machine learning operates in practice.

In all of these settings, data is generated through interaction rather than drawn from a fixed distribution. A learner's own actions or current state determine what feedback and data become

available to it, making the learner's information environment *endogenous*. This endogeneity is a direct consequence of interactivity: when the learner is embedded in an ongoing process with users and other learners, its information can no longer be treated as exogenous. This dissertation studies two feedback loops through which interactive deployment induces endogeneity:

1. **History-dependent feedback in repeated interaction.** When a learner takes actions sequentially, such as recommending items, selecting interventions, or constructing a set of choices over time, the value of each future action depends on what has already been selected. The learner's past actions shape the structure of its own future feedback.

2. **Choice-driven data allocation in multi-learner markets.** When multiple platforms compete for the same pool of users, each user's choice of platform determines which learner observes their data. A learner's current model quality shapes who provides it data, creating a feedback loop between performance and the data distribution.

A concrete setting illustrates both channels. Consider a movie recommendation platform that serves users over time. As it recommends items sequentially, the marginal value of each new recommendation depends on what has already been shown: a sequel gains value if the original was recommended; repeated exposure to similar genres exhibits saturation; some item combinations have complementary effects that neither has alone. The resulting utility is combinatorial and history-dependent, with both diminishing-returns and complementary relationships that purely submodular models cannot capture [20, 77]. A system that ignores this structure and treats each recommendation independently will tend to suggest redundant items from the same genre or miss valuable combinations, because it has no way to account for how past selections change the value of future ones. Now suppose multiple such platforms compete for the same population of users, who choose where to engage based on predictive quality, familiarity, or brand affinity. The data each platform collects becomes a function of its own performance: a platform that serves one audience well attracts more of those users, reinforcing its strength on that niche while leaving it unable to learn about users it fails to attract [27, 60]. This self-reinforcing dynamic, the *overspecialization trap*, can lead to arbitrarily poor global performance even when good models exist in principle. Once a platform falls behind on a subpopulation, the users it loses take their data with them, making recovery progressively harder.

These two forms of endogeneity are not specific to recommendation. History-dependent feedback arises whenever an agent acts sequentially in an environment and learns from the consequences: a language model agent that calls tools, browses the web, and reasons about intermediate results faces exactly this structure [88, 111]. Choice-driven data allocation arises whenever multiple learning systems compete for the same pool of users. Competing language model services are a direct instance: users choose a provider based on quality and preference, and each provider's training data is shaped by who chooses to use it [26]. LLM routing systems, which direct user queries to specialized models based on predicted quality, create the same feedback loop [79]. More broadly, the growing practice of training models on each other's outputs [24, 49, 101] couples the data distributions of multiple learners, with the risk of iterative quality degradation when the process is not carefully managed [95]. As machine learning moves toward more autonomous, multi-agent deployment, these feedback structures will become more prevalent, not less.

Sequential decision-making frameworks such as bandits and reinforcement learning have long studied settings where a learner's actions affect future observations [20, 96], and a growing literature on performative prediction, strategic classification, and multi-agent learning addresses endogenous distribution shift [31, 44, 82, 98]. The specific forms of endogeneity studied in this dissertation, however, fall outside these standard formulations. In the first setting, the core challenge is not sequential decision-making per se, but the *combinatorial structure of the objective*: the utility function exhibits both submodular and supermodular interactions that existing online methods do not handle. In the second setting, the challenge is that the information barrier imposed by user choice *cannot be resolved through exploration alone*: a learner fundamentally cannot observe users who choose a competitor, regardless of its exploration strategy. Overcoming this barrier requires a structurally different mechanism, which motivates the peer probing algorithm developed in this thesis.

This dissertation develops algorithms with provable guarantees for both channels of endogeneity. Chapter 2 extends Gaussian Process contextual bandits to combinatorial objectives with non-submodular structure, integrating Nyström sketching for scalability. Chapter 3 analyzes multi-learner streaming gradient dynamics via stochastic approximation theory, characterizes the overspecialization trap, and proposes peer probing, inspired by knowledge distillation, as a remedy with convergence and performance guarantees. The remainder of this introduction summarizes each chapter's contributions.

1.1 Contributions of Chapter 2

Chapter 2 studies simultaneous online optimization of m related unknown combinatorial objective functions $\{h_1, \ldots, h_m\}$. The learner operates over T time steps, during which it sequentially builds m context-dependent sets $\{S_1, \ldots, S_m\}$ by selecting items from a finite ground set V. At each time step t, a function $h_{u_t} \in \{h_1, \ldots, h_m\}$ arrives. Then, the learner selects an action $v_t \in V$ to add to the corresponding set, and receives noisy marginal-gain feedback y_t, which depends on both v_t and the items selected so far. The utility functions $h_k : 2^V \to \mathbb{R}$ are learned from this feedback over the rounds. The goal is to minimize α-regret: the gap between the learner's cumulative utility and that of an offline greedy algorithm baseline, which possesses full knowledge of the functions. The main contributions are as follows:

1. **Online BP Functions.** We consider utility functions that decompose as $h = f + g$, where f is monotone submodular and g is monotone supermodular (a "BP" decomposition). We prove sublinear α-regret bounds, where α depends on the submodular and supermodular curvatures of the component functions.

2. **Separate Feedback Framework.** We introduce a novel setting where the learner receives separate reward signals for the submodular and supermodular components. This models applications where different aspects of utility (e.g., user satisfaction vs. network effects) can be measured independently. We show that the richer separate feedback enables stronger regret guarantees.

3. **Online Weakly Submodular Functions.** For non-decomposable utility functions, we consider the class of weakly submodular functions, parameterized by a submodularity ratio $\gamma \in [0, 1]$. We prove sublinear α-regret bounds for this broader class.

4. **Scalable Computation via Nyström Sketching.** We address the computational bottleneck of Gaussian Process methods by integrating Nyström approximations, reducing complexity from $O(T^3)$ to $O(TN^2)$ where $N \ll T$ is the sketch size.

5. **Empirical Validation.** We demonstrate our methods on movie recommendation and training data subset selection tasks.

Chapter Outline. Section 2.3 formalizes the problem setting and defines the function classes. Section 2.4 establishes robustness properties of greedy algorithms that underpin our online analysis. Section 2.5 presents our main algorithm and regret bounds for the single (monolithic) feedback setting. Section 2.6 extends to the separate feedback case. Section 2.7 provides numerical experiments.

1.2 Contributions of Chapter 3

Chapter 3 studies online learning dynamics in markets where multiple platforms compete for users. Consider m learners, each maintaining a model θ_i, serving a population distributed according to $\mathscr{P}$ over covariates and labels. Users have *inherent preferences* $\pi(z)$ over platforms (capturing brand loyalty, familiarity, or habits) but also consider predictive quality when choosing. Under a mixture selection rule, users follow their preference with probability τ and otherwise select the platform minimizing their loss. Learners update via streaming gradient descent on their observed users, but critically, the data distribution each learner observes depends on all learners' current models, creating coupled, endogenous dynamics. The goal is to minimize *full-population risk* $\mathscr{R}(\theta) = \mathbb{E}_{z\sim\mathscr{P}}[\ell(z;\theta)]$, not just loss on the observed subpopulation.

Prior work on multi-learner dynamics focuses on convergence to stationary points of aggregate local losses, without analyzing whether these equilibria generalize to unobserved users. We show that standard dynamics can trap learners in overspecialized equilibria, and propose peer probing as a remedy. The main contributions are as follows:

1. **The Failure of Standard Learning.** We analyze Multi-learner Streaming Gradient Descent (MSGD) and prove that due to user selection, MSGD can converge to "bad" stationary points where learners achieve low local loss but arbitrarily poor full-population performance.

2. **Convergence of Peer Probing.** We propose MSGD with Probing (MSGD-P), where learners augment organic user gradients with pseudo-labeled queries sent to peer models. We prove that this multi-agent dynamic converges to a stationary point of a modified potential function. To our knowledge, this is the first analysis of multi-agent dynamics arising from synthetic data training.

3. **Restoring Global Competence.** We characterize the stationary points of MSGD-P and

derive bounds on full-population loss under informational conditions on the probing sources (e.g., probing a known market leader vs. aggregating diverse peers).

4. **Empirical Validation.** We validate our findings on the MovieLens and US Census datasets, showing that peer probing closes the performance gap left by standard learning.

Chapter Outline. Section 3.2 formalizes the market model, user selection rule, and learning objectives. Section 3.3 analyzes standard MSGD dynamics and establishes the existence of over-specialized equilibria. Section 3.4 introduces the peer probing algorithm and proves convergence. Sections 3.4.3 and 3.4.4 characterize when probing restores global competence. Section 3.5 provides empirical validation.

Chapter 2

Online Optimization of Weakly Submodular and BP Functions

As discussed in Chapter 1, interactive learning settings—where a learner repeatedly interacts with an environment over time—are increasingly important in applications such as recommender systems, personalized medicine, and advertisement placement.

The mathematical framework adopted in this chapter is that of *Gaussian Process Contextual Bandits* (GPCB) [17, 63, 91, 96, 102]. In this setting, an agent observes a context ϕ_{u_t} (indexing one of m possible contexts at time t), selects an action $v_t \in V$, and receives a noisy reward. A Gaussian process model provides posterior mean and variance estimates for the reward, which are combined in an Upper Confidence Bound (UCB) rule to balance exploration against exploitation. The standard performance metric is cumulative regret,

$$\mathcal{R}(T) = \sum_{t=1}^{T} f_{\phi_{u_t}}(v^*_{\phi_{u_t}}) - f_{\phi_{u_t}}(v_t),$$

comparing the algorithm's choices against the best action in hindsight for each context.

Chen et al. [20] extended GPCBs to online *combinatorial* optimization, where the agent incrementally constructs a set over time rather than choosing a single action per round. Since offline submodular maximization is NP-hard yet admits an $\alpha = 1 - 1/e$ greedy approximation [77], Chen et al. adopt α-regret, comparing the online algorithm's accumulated utility against the α-approximate offline solution. Unlike standard pointwise regret, this combinatorial notion captures the interde-

	Offline	Pure Online	Online + Nyström	Online + Sep. FB	Online + Nyström + Sep. FB
Modular	N/A	[96] [63]	[113]	N/A	N/A
SM	[77]	[20]	✓	N/A	N/A
BP	[5]	✓	✓	✓	✓
WS	[29] [11]	✓	✓	N/A	N/A

Table 2.1: This chapter's contributions (green ✓ which means new algorithms for sublinear regret) in the context of previous work. Here **SM** refers to SubModular, **BP** to suBmodular-suPermodular, and **WS** to Weakly Submodular. **Sep FB** refers to the separate feedback BP setting introduced in this chapter. N/A means not applicable.

pendencies between elements in the growing set: the function f is unavailable to the algorithm, which observes only noisy marginal gains $y_t = f(v|S_t) + \epsilon_t$ after each selection is committed. The history-dependent nature of this feedback—where past selections shape both future marginal gains and the information available for estimation—makes the online setting substantially more challenging than its offline counterpart and motivates the smoothness assumptions (Assumption 2.1) employed in this chapter.

Despite its many benefits, the purely submodular assumption is not sufficiently expressive to capture essential properties of many real-world environments. As discussed in Chapter 1, user preferences often exhibit complementary (supermodular) relationships—such as movie sequels or synergistic drug combinations—that submodular functions cannot represent.

This chapter establishes sublinear α-regret in the GPCB setting for BP, weakly submodular, and Nyström-accelerated variants of these function classes—none of which had previously been studied in the online combinatorial setting. The remainder of the chapter is organized as follows. Section 2.1 collects the definitions of submodularity, supermodularity, BP functions, and the associated curvature parameters. Section 2.2 surveys related work. Section 2.3 formalizes the problem setting. Section 2.4 establishes the robustness of the offline greedy algorithm to approximate selections, which is of independent interest. Section 2.5 develops the MNN-UCB algorithm and proves sublinear α-regret for BP and weakly submodular functions under monolithic feedback. Section 2.6 shows that separate feedback yields stronger guarantees via a distorted greedy approach. Section 2.7 presents numerical experiments on movie recommendations and training-data subset selection.

2.1 Preliminaries

A set function $h : 2^V \to \mathbb{R}$ maps any subset of a finite ground set V of size $|V| = n$ to the reals. Arbitrary set functions are impossible to optimize with any quality assurance guarantee without an exponential cost, so we restrict attention to set functions with useful structural properties.

A set function $f : 2^V \to \mathbb{R}$ is said to be monotone non-decreasing if $f(A \cup \{v\}) \geq f(A)$ for all $A \subseteq V, v \in V$. It is normalized if $f(\emptyset) = 0$. For convenience, we refer to the collection of **Monotone Non-decreasing Normalized** set functions as **MNN** functions. We use the gain notation $f(v|S) = f(S \cup \{v\}) - f(S)$ to denote the **marginal gain** of adding element v to the set S.

A set function f defined over the ground set V is called **submodular** if for all $A \subseteq B \subseteq V$ and any element $v \notin B$ we have $f(A \cup \{v\}) - f(A) \geq f(B \cup \{v\}) - f(B)$. A function $g : 2^V \to R$ is said to be **supermodular** if $-g$ is submodular — g has the property of *increasing returns* where the presence of an item can only enhance the utility of selecting another item. The class of functions defined below is the primary focus of this chapter.

Definition 2.1 (BP Function)**.** *A utility function h is said to be BP if it admits the decomposition $h = f + g$, where f is submodular, g is supermodular, and both functions are also MNN.*

Next, we introduce the notion of curvature for submodular and supermodular functions. This enables us to state the assumptions required to obtain approximation bounds for offline BP functions, as established by [5].

Definition 2.2 (Submodular curvature)**.** *Denote the curvature for submodular f as $\kappa_f = 1 - \min_{v \in V} \frac{f(v|V \setminus \{v\})}{f(v)}$.*

Definition 2.3 (Supermodular curvature)**.** *Denote the curvature for supermodular g as: $\kappa^g = 1 - \min_{v \in V} \frac{g(v)}{g(v|V \setminus \{v\})}$.*

These quantities are contained in $[0, 1]$ and measure how far the functions are from being modular: if a curvature is zero, the function is modular. Given the function, these can be calculated in time linear in $|V|$. Bai and Bilmes [5] analyzed the greedy algorithm for the cardinality-constrained BP maximization problem and provided a $\frac{1}{\kappa_f}\left[1 - e^{-(1-\kappa^g)\kappa_f}\right]$ approximation ratio. They also showed that not all monotone non-decreasing set functions admit a BP decomposition. However, in cases

where such a decomposition is available, one can easily compute the curvature of submodular and supermodular terms and compute the bound.

Since not all MNN functions are representable as BP functions, we also study arbitrary MNN functions in terms of how far they are from being submodular.

Definition 2.4 (Submodularity ratio, [11, 29, 30])**.** *The submodularity ratio of a non-negative set function $h(\cdot)$ is the largest scalar γ such that $\sum_{v\in S\setminus A} h(v|A) \geq \gamma h(S|A), \forall S, A \subseteq V$.*

The submodularity ratio measures to what extent $h(\cdot)$ has submodular properties. For a non-decreasing function $h(\cdot)$, it holds that $\gamma \in [0, 1]$ always, and $h(\cdot)$ is submodular if and only if $\gamma = 1$.

Definition 2.5 (Generalized curvature, [11])**.** *The curvature of a non-negative function $h(\cdot)$ is the smallest scalar ζ such that $\forall S, A \subseteq V, v \in A\setminus S$, $h(v|A\setminus\{v\} \cup S)) \geq (1-\zeta)h(v|A\setminus\{v\})$.*

Unlike the submodular and supermodular curvatures, the submodularity ratio and generalized curvature are information-theoretically hard to compute in general [5]. We refer to MNN set functions with bounded submodularity ratio γ and generalized curvature ζ as **weakly submodular** (WS). Bian et al. [11] analyzed the greedy algorithm for maximizing such functions subject to a cardinality constraint and obtained a $\frac{1}{\zeta}\left(1 - e^{-\zeta\gamma}\right)$ approximation ratio.

2.2 Related Work

Submodular maximization with bounded curvature. Nemhauser et al. [77] studied the greedy algorithm for maximizing a monotone non-decreasing submodular set function subject to a cardinality constraint and provided a $1 - \frac{1}{e}$ approximation ratio. While Nemhauser and Wolsey [76] showed that this factor cannot be improved under a polynomial number of function value queries, the greedy algorithm usually performs closer to optimal in practice. To quantify this, Conforti and Cornuéjols [28] introduced the notion of *curvature* $\kappa \in [0, 1]$ for submodular functions (Definition 2.2) and showed that the greedy algorithm achieves a $\frac{1}{\kappa}(1 - e^{-\kappa})$ approximation ratio. For general submodular functions ($\kappa = 1$), this recovers the $1 - \frac{1}{e}$ bound; as $\kappa \to 0$, the ratio tends to 1. More recently, Sviridenko et al. [99] obtained a $1 - \frac{\kappa}{e}$ approximation ratio for the more general matroid

constraint setting, with matching lower bounds showing optimality. The notion of curvature has since been extended to continuous submodular functions [85–87, 92].

BP maximization. Bai and Bilmes [5] introduced the problem of maximizing a BP function $h = f + g$ (Definition 2.1) subject to a cardinality constraint as well as the intersection-of-p-matroids constraint. They showed that this problem is NP-hard to approximate to any factor without further assumptions. However, if the supermodular function g has bounded curvature ($\kappa^g < 1$), they analyzed the greedy algorithm and provided a $\frac{1}{\kappa_f}\big(1 - e^{-(1-\kappa^g)\kappa_f}\big)$ approximation ratio. For general supermodular functions ($\kappa^g = 1$), the ratio is 0; as $\kappa^g \to 0$, the bound recovers that of Conforti and Cornuéjols [28]. More recently, Liu et al. [70] proposed a distorted version of the greedy algorithm with an improved $\min\{1 - \frac{\kappa_f}{e}, 1 - \kappa^g e^{-(1-\kappa^g)}\}$ approximation ratio.

Submodularity ratio. Das and Kempe [29] introduced the submodularity ratio γ and generalized curvature ζ for general monotone non-decreasing set functions (Definitions 2.4 and 2.5) and showed that the greedy algorithm obtains a $\frac{1}{\zeta}\big(1 - e^{-\zeta\gamma}\big)$ approximation ratio under cardinality constraints [11, 29]. Unlike the BP decomposition, these parameters can be defined for any monotone non-decreasing set function but are, in general, exponentially costly to compute.

Adaptive and interactive submodularity. Chen et al. [20] is the work most closely related to this chapter—they employ a similar UCB algorithm to optimize an unknown submodular function in an interactive setting. They define regret as the sub-optimality gap with respect to a full-knowledge greedy strategy at the final round, and relate it to a pointwise regret that accumulates stage-by-stage. By viewing the submodular problem as a special case of contextual bandits, they bound this accumulation using the kernel bandit results of [63]. Golovin and Krause [38], Guillory and Bilmes [42] also consider adaptive or interactive submodular problems under stronger structural assumptions.

Kernel bandits. Srinivas et al. [96] consider the problem of optimizing an unknown function f that is either sampled from a Gaussian process or has bounded RKHS norm. They develop GP-UCB, an upper-confidence bound approach that achieves sublinear regret depending on an information-gain term γ_T. Krause and Ong [63] extend this to the contextual setting where the function f_{z_t} depends

on a time-varying context z_t. Valko et al. [102] replace the γ_T scaling with $\sqrt{\gamma_T}$, and Camilleri et al. [17] extend experimental design for linear bandits to the kernel setting with batch support. Zenati et al. [113] use Nyström points to speed up the algorithm with the same asymptotic regret guarantee as GP-UCB, which directly inspires the algorithms developed in this chapter.

Combinatorial bandits. In [22, 64, 81, 100], the optimizer chooses a set at each time step and the submodularity is between elements chosen in the single time step. In this chapter, the optimizer chooses a single item at each time and accumulates a set over time; the submodularity is between elements chosen at different time steps. While the formulations appear similar, they apply to different settings.

Comparison with [21, 69]. These papers have titles similar to this chapter but address a different setting, better described as "streaming" rather than "online" [4, 19, 34]. The approach in [69] assumes the function h is known with arbitrary queries available and no cost for evaluating it. Items are revealed one by one in a fixed order, and the algorithm must decide whether to add each item to the set or discard it forever. There is no statistical estimation component, and competitive ratio bounds are provided rather than regret bounds.

2.3 Problem Formulation

The optimizer operates in an environment that occurs over T time steps. Specifically, at each time step $t \in [T]$:

1. The optimizer encounters one of m set functions from the set $\{h_1 \dots h_m\}$ each defined over the finite ground set V. The optimizer is ignorant of the function but knows its index $u_t \in [m]$ as well as a context or feature-vector ϕ_{u_t} for that index at round t.

2. The optimizer computes and then performs/plays action $v_t \in V$, and then adds v_t to its growing context-dependent set $S_{t_{u_t},u_t}$ of size $|S_{t_{u_t},u_t}| = t_{u_t}$ with $\sum_{j\in[m]} t_j = t$. The set $S_{t_{u_t},u_t}$ contains all items so far selected for the unknown function h_{u_t}.

3. The environment offers the optimizer noisy marginal gain feedback. There are two feedback models:

(3a) *Monolithic Feedback:* The optimizer receives y_t with $y_t = h_{u_t}(v_t|S_{t_{u_t},u_t}) + \epsilon_t$.

(3b) *Separate Feedback:* In the BP case, pair $(y_{f,t}, y_{g,t})$ may be available with $y_{f,t} = f_{u_t}(v_t|S_{t_{u_t},u_t}) + \epsilon_t/2$ and $y_{g,t} = g_{u_t}(v_t|S_{t_{u_t},u_t}) + \epsilon_t/2$.

The separate feedback case (3b) is relevant only for applications (e.g., multiple surveys, etc.) where it is feasible. Section 2.6 exploits this richer feedback to improve performance. All feature-vectors ϕ_{u_t} are chosen from set Φ of size $|\Phi| = m$, and we assume that the identity of the utility function h_q is determined uniquely by ϕ_q; hence, when clear from context, we use h_q to refer to h_{ϕ_q}.

Two applications illustrate how this framework may be instantiated. Vignette 2.2 is further explored in Appendix A.6.

Vignette 2.1 (Movie Recommendations)**.** Each function h_q captures the preferences of a single user $q \in [m]$, and the index $u_t \in [m]$ reveals which user has arrived at time step t. The action v_t performed at time t is the optimizer's recommended movie to user u_t. The feature vector ϕ_{u_t} contains user-specific information, e.g., age range, favorite movies and genres, etc. The feedback gain $h_{u_t}(v|A)$ is the enjoyment user u_t has from watching movie v having already watched the movies in set A.

Vignette 2.2 (Active Learning)**.** The optimizer chooses training points to be labeled for m related tasks on the same dataset - for instance classification, object detection, and captioning. The function $h_q(A)$ is the test accuracy of a classifier $f(A)$ trained on set A on the q^{th} task. Choosing an action v_t is tantamount to choosing a training point to be labeled for task $u_t \in [m]$.

Table 2.1 summarizes the results established in this chapter for different function classes and feedback models, alongside the prior work each result generalizes.

To design low-regret online item-selection strategies for these problems (made precise in Section 2.5), the first step is to study the robustness of the greedy procedure for the offline optimization of Monotone Non-decreasing Normalized (MNN) functions (see Section 2.1) in Section 2.4. Then Section 2.5 shows that the proposed online procedure approximates the offline greedy algorithm, leveraging Section 2.4 to obtain online guarantees.

2.4 Offline Algorithm Robustness

We consider the problem of cardinality-constrained optimization of a MNN objective $h : 2^V \to \mathbb{R}$.

$$\max_{S \in 2^V} h(S) : |S| \leq k. \tag{2.1}$$

Let S^* denote an achieving set solving Equation (2.1). The most common approximation algorithm for this problem greedily [77] maximizes the available marginal gain having oracle access to h. In online settings, however, this oracle access is not available. To analyze the online setting, we consider a modified offline algorithm where the greedy choices might be good only with respect to a set of additive "slack" variables r_j, exploring the impact of this modification on approximation quality for different classes of functions. Then in Section 2.5 we develop an online algorithm that emulates greedy in this way.

2.4.1 Greedy Selection Robustness

We define an **approximate greedy** selection rule that, given scalars $\{r_j\}_{j=1}^k$, chooses v_j for each $j \in [k]$ satisfying

$$v_j \in \{v : h(v|S_{j-1}) \geq \text{argmax}_{\tilde{v}}\, h(\tilde{v}|S_{j-1}) - r_j\}, \tag{2.2}$$

where $S_j = \{v_1 \dots v_j\}$ and S^* the optimal set of size k.

Lemma 2.1. *Any output S of the approximate greedy selection rule in Equation (2.2) admits the following guarantee for BP objectives (Def. 2.1) for Problem (2.1):*

$$h(S) \geq \frac{1}{\kappa_f}\left[1 - e^{-(1-\kappa^g)\kappa_f}\right] h(S^*) - \sum_{j=1}^{k} r_j,$$

where κ_f, κ^g are as defined in Definitions 2.2 and 2.3.

This result is a generalization of Bai and Bilmes [5, Theorem 3.7] which is recovered by setting $\forall j, r_j = 0$. This result is surprising because, with the supermodular part of the BP function, poor early selections may preclude the ability to exploit potential increasing returns from g — the curvature κ^g is crucial for this. The result can also be understood as a generalization of Chen et al. [20], which studies the robustness of the greedy algorithm to errors in submodular functions. In

their case, however, they adapt the simple classical greedy algorithm proof [77]. In Appendix A.3, we provide an alternate proof using a crude bound that incorporates the supermodular curvature but ignores the submodular curvature, reminiscent of the argument in Chen et al. [20]. However, the approximation ratio obtained is much worse than that of Bai and Bilmes [5].

The proof below uses the detailed analysis in Bai and Bilmes [5]. This poses a considerable challenge, since Bai and Bilmes [5] (inspired by Conforti and Cornuéjols [28]) formulate an intricately designed series of linear programs to show that any selection that has as much overlap with the optimal solution as the greedy algorithm must achieve the desired approximation ratio. Here, the errors r_j manifest as perturbations to the constraints of the linear programs. We then perform a *sensitivity analysis* of the linear programs to argue that these perturbations to the constraints lead to a linear perturbation to the optimal objective and does not cause it to explode.

We use S_t to refer to the ordered set of elements chosen for function h until round t, and S to refer to the ordered final set of items chosen for function h until round T. Hence, S_j refers to the first j elements chosen for h. Let s_j be the j^{th} element of S. Then, we define $a_j = h(s_j|\{s_1 \dots s_{j-1}\})$ be the gain of the j^{th} element chosen.

Recall that S is an ordered set. We let $C \subseteq [k]$ denote the indices (in increasing order) of elements in S that are also in S^*. For instance, for $S = \{s_1 \dots s_5\}$ and $S \cap S^* = \{s_1, s_2, s_3\}$, we have $C = \{1, 2, 3\}$. Hence, $j \in C \iff s_j \in S \cap S^*$. Further, define filtered sets $C_t = \{c \in C | c \leq t\}$ as the subset of the first t_{th} elements of S that are also in the optimal S^*.

Proof of Lemma 2.1. From Lemma A.2 (in Appendix A.4.1), we have that the approximate greedy procedure obeys k different inequalities, and we wish to show that this is sufficient to obey the inequality above. In order to complete the argument, we consider the worst-case overall gain if these k inequalities are satisfied; and show that this worst-case sequence satisfies the desired, and hence the approximate greedy procedure must satisfy the desired as well.

To characterize the worst-case gains, we define a set of linear programming problems parameterized

by a set B and constants (ξ, ρ).

$$\begin{aligned} T(B, \xi, \rho) = \min_{b} & \sum_{j=1}^{k} b_j \\ \text{s.t.} \quad h(S^*) \leq \xi & \sum_{j \in [t-1] \setminus B_{t-1}} b_j + \sum_{j \in B_{t-1}} b_j + \frac{k - |B_{t-1}|}{1-\beta} b_t \quad , \forall t \in [k]. \end{aligned} \tag{2.3}$$

In the above, the decision variable $b = [b_1 \dots b_k]$ is a vector in $\mathbb{R}^k$, and satisfies $b \geq 0$. The constants k is a fixed value for the LP. The parameter of the LP, $B \subseteq [k]$, and $B_t = \{j \in B | j \leq t\}$ is the filtered set. Note that the constraints are linear in b with non-negative coefficients.

The above LP becomes helpful to our setting when we set $(\xi, \beta) = (\kappa_f, \kappa^g)$. Additionally, we are interested in the choices $B = C$ and $B = \emptyset$, where C is defined prior to the lemma statement. To show the result, we hope to show the following chain of inequalities:

$$h(S) + \sum_{j=1}^{k} r_j \geq T(C, \kappa_f, \kappa^g) \geq T(\emptyset, \kappa_f, \kappa^g) \geq \omega h(S^*). \tag{2.4}$$

In the above,

$$\omega = \frac{1}{\kappa_f} \left[1 - e^{-(1-\kappa^g)\kappa_f} \right].$$

Combining the two ends of this chain yields the desired lemma statement. We recognize that $T(\cdot)$ is exactly the LP considered in [5], modulo notation differences. Since the second and third inequality are just statements about the linear program, they follow directly from Lemma D.2 in [5] when we substitute $\xi = \kappa_f$ and $\beta = \kappa^g$.

For the first inequality, we have from Lemma A.2 that $b_j = a_j + r_j$ is a feasible solution for the linear program $T(C, \kappa_f, \kappa^g)$. Hence,

$$T(C, \kappa_f, \kappa^g) \leq \sum_{j=1}^{k} b_j = \sum_{j=1}^{k} a_j + \sum_{j=1}^{k} r_j = h(S) + \sum_{j=1}^{k} r_j.$$

□

In the case where h does not have a BP decomposition, we offer the following result generalizing Bian et al. [11].

Lemma 2.2. *Any output S of the approximate greedy selection rule in Equation (2.2) admits the following guarantee on objectives with submodularity ratio γ and generalized curvature ζ (Definitions 2.4 and 2.5) for Problem (2.1):*

$$h(S) \geq \frac{1}{\zeta}\left(1 - e^{-\zeta\gamma}\right) h(S^*) - \sum_{j=1}^{k} r_j.$$

We see in Section 2.5 that Lemmas 2.1 and 2.2 are key to the analysis of Algorithm 1. The proof follows the same LP construction as Lemma 2.1, but with different constants reflecting the weakly submodular function class. Define S, s_j, a_j, C as in the proof of Lemma 2.1.

Proof of Lemma 2.2. We consider again the parameterized LP $T(\cdot)$, but this time with the constants set as $\xi = \zeta, \rho = 1 - \gamma$. To show the result, we hope to show the following chain of inequalities:

$$h(S) + \sum_{j=1}^{k} r_j \geq T(C, \zeta, 1-\gamma) \geq T(\phi, \zeta, 1-\gamma) \geq \omega h(S^*). \tag{2.5}$$

In the above,

$$\omega = \frac{1}{\zeta}\left[1 - \left(1 - \frac{\gamma\zeta}{k}\right)^k\right].$$

Similarly to the argument in Lemma 2.1, the first two inequalities follow directly from Lemma D.2 in [5] when we substitute $\xi = \zeta$ and $\rho = 1 - \gamma$. Under the same choice of constants, we have from Lemma A.3 (in Appendix A.4.2) that $b_j = a_j + r_j$ is a feasible solution for the linear program $T(C, \zeta, 1-\gamma)$. Hence,

$$T(C, \zeta, 1-\gamma) \leq \sum_{j=1}^{k} b_j = \sum_{j=1}^{k} a_j + \sum_{j=1}^{k} r_j = h(S) + \sum_{j=1}^{k} r_j.$$

Recognizing that

$$\frac{1}{\zeta}\left[1 - \left(1 - \frac{\gamma\zeta}{k}\right)^k\right] \geq 1 - e^{-\zeta\gamma}.$$

completes the argument. □

2.4.2 Distorted BP Greedy Robustness

In Liu et al. [70], the authors present a "distorted" version of the greedy algorithm, which achieves a better greedy approximation ratio than Bai and Bilmes [5] for Problem (2.1) with a BP objective.

Here, we study its robustness.

As in Sviridenko et al. [99], we define the modular lower bound of the submodular function $l_1(S) = \sum_{j \in S} f(j|V \backslash \{j\})$. Also, define the totally normalized submodular function as $f_1(S) = f(S) - l_1(S)$. Note that f_1 always has curvature $\kappa_f = 1$ and also that $h(S) = f_1(S) + g(S) + l_1(S)$. We define the function $\pi_j(v|A)$ as follows:

$$\pi_j(v|A) = \left(1 - \frac{1}{k}\right)^{k-j-1} f_1(v|A) + g(v|A) + l_1(v). \tag{2.6}$$

In Liu et al. [70], the optimizer greedily maximizes the π_j function at step j rather than the original BP gain. In π_j, the submodular part is down weighted relative to the supermodular part. Intuitively, this is helpful because the supermodular part is initially much smaller than the submodular part, but ultimately dominates the sum. Thus, it is in the optimizer's interest to focus on the supermodular part early, rather than waiting until it becomes large.

We define the **approximate distorted greedy** selection rule as follows. Given scalars $\{r_j\}_{j=1}^k$, in each step $j = \{1, \ldots, k\}$, the optimizer chooses an item v_j that satisfies

$$v_j \in \{v : \pi_j(v|S_{j-1}) \geq \operatorname{argmax}_{\tilde{v}} \pi_j(\tilde{v}|S_{j-1}) - r_j\}. \tag{2.7}$$

We present a robust version of Liu et al. [70]:

Lemma 2.3. *Any output S of the approximate distorted greedy selection rule in Equation (2.7) admits the following guarantee for Problem (2.1) with a BP objective (Def. 2.1):*

$$h(S) \geq \min\left\{1 - \frac{\kappa_f}{e},\ 1 - \kappa^g\right\} h(S^*) - \sum_{j=1}^{k} r_j,$$

where κ_f, κ^g are as defined in Def. 2.2 and 2.3.

This lemma is the key to the analysis of Algorithm 4 in Section 2.6. We remark that the approximation ratio above is different from Liu et al. [70]. This is due to a correction of an error in their analysis, which caused the approximation ratio to change from their $\alpha = \min\left\{1 - \frac{\kappa_{f,q}}{e},\ 1 - \kappa_q^g e^{(1-\kappa_q^g)}\right\}$ to the bound above. Details are in Appendix A.5.5. Additionally, note that Sviridenko et al. [99] provided a $1 - \frac{\kappa_f}{e}$ lower bound for monotone submodular maximization and later on, Bai and Bilmes [5] obtained

a $1 - \kappa^g$ lower bound for monotone supermodular maximization. The corrected approximation ratio in Equation (2.15) is simply the minimum of these two quantities. In Appendix A.5, we provide a heat map that compares this approximation ratio to that of Bai and Bilmes [5], showing that it is strictly greater for all κ_f, κ^g. Once their analysis is fixed, we adapt their argument to the more general case that allows for errors r_j at each stage to complete the robust online proofs. We also define the modular lower bound of the supermodular function $l_2(S) = \sum_{j \in S} g(j|\emptyset)$, the totally normalized supermodular function $g_1(S) = g(S) - l_2(S)$, and $l = l_1 + l_2$.

Proof of Lemma 2.3. Using the submodular and supermodular curvature definition, we can write

$$l_1(S) = \sum_{j \in S} f(j|V \backslash \{j\}) \geq (1 - \kappa_f) f(S)$$
$$l_2(S) = \sum_{j \in S} g(j|\emptyset) \geq (1 - \kappa^g) g(S)$$

Then, we can use the result of Lemma A.5 (in Appendix A.4.3) to write

$$\begin{aligned}
f(S) + g(S) &= f_1(S) + g_1(S) + l(S) \\
&\geq \left(1 - \frac{1}{e}\right) f_1(S^*) + l(S^*) - \sum_{j=1}^{k} r_j \\
&= \left(1 - \frac{1}{e}\right) (f(S^*) - l_1(S^*)) + l_1(S^*) + l_2(S^*) - \sum_{j=1}^{k} r_j \\
&= \left(1 - \frac{1}{e}\right) f(S^*) + \frac{1}{e} l_1(S^*) + l_2(S^*) - \sum_{j=1}^{k} r_j \\
&\geq \left(1 - \frac{1}{e}\right) f(S^*) + \frac{1 - \kappa_f}{e} f(S^*) + (1 - \kappa^g) g(S^*) - \sum_{j=1}^{k} r_j \\
&= \left(1 - \frac{\kappa_f}{e}\right) f(S^*) + (1 - \kappa^g) g(S^*) - \sum_{j=1}^{k} r_j \\
&\geq \min\left\{1 - \frac{\kappa_f}{e},\ 1 - \kappa^g\right\} h_q(S^*) - \sum_{j=1}^{k} r_j.
\end{aligned}$$

□

2.5 No-Regret Single Feedback

In the previous section, we considered the robustness of the greedy algorithm in the offline setting. This section returns to the interactive problem from Section 2.3 and will show how it reduces to the offline problem.

First, the notion of scaled regret mentioned in the introduction to this chapter is fully defined. The scaling compares with the appropriate offline algorithm for the relevant function class; it is standard to consider scaled regret for NP-hard problems (e.g., [20]). Recall the interactive setup from Section 2.3. Let T_q represent the number of items selected for function h_q by the final round, T, so that $\sum_{q=1}^{m} T_q = T$. The set $S_{T_q,q}$ is the final selection for h_q and we set $S_q = S_{T_q,q}$ for notational simplicity. Let $S_q^* \in \operatorname{argmax}_{|S|\leq T_q} h_q(S)$ be a maximizing payoff set for h_q with at most T_q elements. Inspired by Bai and Bilmes [5] and with respect to the approximation ratio obtained for the greedy baseline for BP functions, we define the regret metric $\mathcal{R}_{\text{BP}}(T)$ as follows:

$$\mathcal{R}_{\text{BP}}(T) := \sum_{q=1}^{m} \frac{1}{\kappa_{q,f}} \left[1 - e^{-(1-\kappa_q^g)\kappa_{q,f}}\right] h_q(S_q^*) - h_q(S_q). \tag{2.8}$$

From Lemma 2.1, if the online algorithm is approximately greedy as in Equation (2.2), then the regret is bounded by the accumulation of the approximation errors r_j. This observation bridges the gap between the online and offline settings. Hence, the goal is to design an algorithm that satisfies these properties. Analogously, for functions with bounded submodularity ratio γ_q (Definition 2.4) and generalized curvature ζ_q (Definition 2.5), we define:

$$\mathcal{R}_{\text{WS}}(T) := \sum_{q=1}^{m} \frac{1}{\zeta_q} \left[1 - e^{-\zeta_q \gamma_q}\right] h_q(S_q^*) - h_q(S_q). \tag{2.9}$$

If we knew all the functions $\{h_1 \dots h_m\}$, we could select the greedy item at each stage and achieve zero regret. Define $\Delta(\phi, S, v) = h_\phi(v|S)$ to encapsulate all m latent objectives succinctly (the notational shortcuts $x_t = (\phi_{u_t}, S_{u_t}, v_t)$ and $\Delta(x_t)$ are also used below). Knowing this function is equivalent to knowing $\{h_1 \dots h_m\}$. Thus, the task is to design a procedure to estimate $\Delta(\phi, v, S)$ from data such that the approximation errors r_j reduce over time.

To make this possible, we must make additional assumptions on $\Delta(\cdot)$. To see why, consider what we can infer from an observation without any additional assumptions. In the BP case, for instance,

the q-th BP gain function is uniquely defined by $2^{|V|}$ function evaluations $h_q(v|S)$ for each possible (v, S). If we observe $h_q(v|S)$ for some (v, S), then we can only make inferences about $f_q(v|A)$ and $g_q(v|A)$ for all $A \subseteq S$ or $A \supseteq S$; since we can only choose item v once during the optimization for user q, this information is not useful practically. This motivates the following assumption.[1]

Assumption 2.1. *The $\Delta(\cdot)$ function lives in a Reproducing Kernel Hilbert Space (RKHS) associated with some kernel $\hbar$ and has bounded norm i.e $\|\Delta\|_{\hbar} \leq B$.*

The assumption ensures the outputs of the $\Delta(\cdot)$ function vary smoothly with respect to the inputs and is standard with GPCBs [20, 63, 91, 96]. E.g., if two related movies are watched by two similar users, they should provide similar ratings. Thus, each query provides information about all $m \cdot 2^{|V|}$ other possible queries to all functions, making estimation feasible since the kernel $\hbar((\phi_q, S, v), (\phi_{q'}, S', v'))$ measures similarity between two inputs.

2.5.1 MNN-UCB Algorithm

Algorithm 1 MNN-UCB

Input set V, kernel function $\hbar$

Initialize $S_q \leftarrow \emptyset, V_q \leftarrow V, \ \forall q \in [m]$
Initialize $X_0 \leftarrow \emptyset, G_1 \leftarrow \emptyset$
for $t \in \{1, 2, 3, \ldots, T\}$ **do**
  Observe u_t from environment.
  if $t = 1$ **then**
    Choose $v_1 \in V_{u_t}$ uniformly at random
  else
    Calculate $\mu_t, \sigma_t =$
      MVCalc$(V_{u_t}, \phi_{u_t}, S_{u_t}, G_t, G_{t-1}, x_{t-1}, X_{t-1}, \mathbf{y}_{t-1})$
    Select $v_t \leftarrow \text{argmax}_{v \in V_{u_t}} \mu_t(v) + \beta_t \sigma_t(v)$
  end if
  Obtain feedback $y_t = \Delta(x_t) + \epsilon_t$
  Update $S_{u_t} \leftarrow S_{u_t} \cup \{v_t\}, V_{u_t} \leftarrow V_{u_t} \setminus v_t$
  Update $x_t \leftarrow (\phi_{u_t}, S_{u_t}, v_t), X_t \leftarrow [x_1, x_2, \ldots x_t]$
  Update $\mathbf{y}_t \leftarrow [y_1, y_2, \ldots, y_t]^\top$
  Decide whether to store new point: $G_{t+1} =$ NystromSelect$(\hbar, G_t, x_t)$
end for

[1] See [10] for a comprehensive treatment of RKHS and kernels.

Algorithm 2 NystromSelect

Input: $\hbar, G_t, x_t$;
Locally stored variables: List L
Hyperparams: Regularization λ, Accuracy η, Budget b

If first call, init L to an empty list.
Compute leverage score $\hat{\tau}_t(\lambda, \eta, x_t)$ from Eq. (2.10)
With probability $\min(b \cdot \hat{\tau}_t(\lambda, \eta, x_t), 1)$ include x_t in G_{t+1}.
Append $\hat{\tau}_t(\lambda, \eta, x_t)$ to L
Result: G_{t+1}

Algorithm 3 MVCalc.

Input: $V_{u_t}, \phi_{u_t}, S_{u_t}, G_t, G_{t-1}, x_{t-1}, S, \mathbf{y}_t$
Locally stored variables: L_1, L_2, L_3
Hyperparams: Regularization λ

// Update $K_{GG}^{-1}, \Lambda_t, \tilde{\mathbf{y}}_t$.
if $|G_t| = 1$ **then**
  Init L_1, L_2, L_3 each to empty lists
  Init $\tilde{\mathbf{y}}_t = y_t \hbar(x_{t-1}, x_{t-1})$
  Init $K_{G_tG_t}^{-1} = 1/\hbar(x_{t-1}, x_{t-1})$
  Init $\Lambda_t = 1/[\hbar(x_{t-1}, x_{t-1})^2 + \lambda \hbar(x_{t-1}, x_{t-1})]$.
else if $G_t = G_{t-1}$ **then**
  Update Λ_t using Eq (2.11)
  // Next line retreives $\tilde{\mathbf{y}}_{t-1}$ from L_1
  $\tilde{\mathbf{y}}_t = \tilde{\mathbf{y}}_{t-1} + y_t \hbar_{G_t}(x_{t-1})$
else
  // Next two lines use lists L_2, L_3 resp.
  Update Λ_t using Eq. (2.11) with Schur complements.
  Update $K_{G_tG_t}^{-1}$ using Eq (2.13) with Schur complements
  $\tilde{\mathbf{y}}_t = [\tilde{\mathbf{y}}_{t-1} + y_t \hbar_{G_t}(x_{t-1}), K_S(x_{t-1})^\top \mathbf{y}_t]^\top$
end if
$z_t \leftarrow (\phi_{u_t}, S_{u_t})$
Append $(\tilde{\mathbf{y}}_t, \Lambda_t, K_{GG}^{-1})$ to L_1, L_2, L_3 lists resp
// Calculate mean and variance vectors.
for $v \in V_{u_t}$ **do**
  $\tilde{\mu}_t(v) \leftarrow \hbar_{G_t}((z_t, v))^T \ \Lambda_t \tilde{\mathbf{y}}_t$
  $\delta_t(v) \leftarrow \hbar_{G_t}((z_t, v))^T (\Lambda_t - \lambda^{-1} \ K_{GG}^{-1}) \hbar_{G_t}((z_t, v))$
  $\tilde{\sigma}_t(v)^2 \leftarrow \lambda^{-1} \hbar((z_t, v), (z_t, v)) + \delta_t(v)$
end for
Result: $\{\tilde{\mu}_t(v)\}_{v \in V_{u_t}}, \{\tilde{\sigma}_t(v)\}_{v \in V_{u_t}}$

Algorithm 1 is inspired by [20, 113] based on Upper Confidence Bound (UCB) algorithms for kernel bandits. At time $t \in [T]$, the optimizer has available the noisy evaluations of the unknown $\Delta(\cdot)$ function in vector $\mathbf{y}_t = (y_j)_{j=1}^{t-1}$ for corresponding inputs held in vector $X_t = (x_j)_{j=1}^{t-1}$ — these are updated at the end of each iteration. These are used by the subroutine MVCalc that, using GP kernel techniques [96, 102] and the Nyström set of samples $G_t \subset X_t$, efficiently compute estimates of

the GP posterior distribution's conditional mean and variance used for the UCB marginal gains in the maximization (line 10 of Algorithm 1).

That is, the algorithm chooses the item v_t that has the highest UCB in line 11 where the parameter β_t controls the algorithm's propensity towards either exploration or exploitation (see Appendix A.5.2). We use the notation $\hbar_A(x) = [\hbar(x_1, x) \dots \hbar(x_{|A|}, x)]$ to measure the similarity between x and every element in $A = \{x_j\}_{j \in \{1,\dots,|A|\}}$. Hence, $\hbar_{G_t}((\phi_{u_t}, S_{u_t}, v))$ measures the similarity of the input (ϕ_{u_t}, S_{u_t}, v) to the historical data in Nyström set G_t. Notation $K_{AB}(v) = [\hbar(x, x')]_{x \in A, x' \in B}$ contains the matrix of pairwise kernel-similarities for elements in A, B and $K_{G_t G_t}$ is the covariance matrix of the historical data G_t. Below, we describe the details for the two subroutines used in Algorithm 1, for the readers who are interested in calculations for kernel updates, and selection of informative points to improve computation via Nyström sampling.

Efficiency and NyströmSelect In prior submodular bandits work [20], each iteration $t \in [T]$ needs to invert a $t \times t$ matrix since *all* historical data X_t is used when calculating the conditional means and variances. Even if online matrix-inverse techniques are used, the run-time becomes $O(T^3)$, which is impractical. We use Nyström sampling to mitigate this and only use a selected subset $G_t \subset X_t$ of historical data to compute $\mu_t(v), \sigma_t(v)$ for all $v \in V_{u_t}$ [113]. Nyström sampling chooses the points that are most useful for prediction. To define this precisely, we introduce a bit of notation. Define $G' = G_t \cup x_t$. Define the estimated leverage score $\hat{\tau}_t(\lambda, \eta, x)$ as:

$$\frac{1+\eta}{\lambda}\left[\hbar(x,x) - \tilde{\hbar}_{G'}(x)(\tilde{K}_{G'G'} + \lambda I)^{-1}\tilde{\hbar}_{G'}(x)\right]. \tag{2.10}$$

Define M_t as follows:

$$M_t = \begin{bmatrix} \operatorname{diag}\left([\min(\hat{\tau}_j(\lambda, \eta, x_j), 1)]_{x_j \in G_t}\right) & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix}.$$

It is the diagonal scaling matrix of (clipped) leverage scores of past selected points with an extra entry with value 1 for the hypothetical new point x_t. In Equation (2.10) above, we define $\tilde{\hbar}_{G'}(x_t) = M_t^\top \hbar_{G'}(x_t)$ and $\tilde{K}_{G'G'} = M_t^\top K_{G'G'} M_t$. The point x_t is included into the Nyström set G_{t+1} in with probability proportional to $\hat{\tau}_t(\lambda, \eta, x_t)$ (line 3 of Algorithm 2). To understand why this is reasonable, note that $\hat{\tau}_t(\lambda, \eta, x_t)$ is shown to estimate the ridge leverage score (RLS) of a point well [15, 16]. The

RLS measures intuitively how correlated the new point is to previous points; if it is highly correlated, it will be sampled with low probability, but if it is orthogonal, it will be sampled with high probability. This procedure improves the runtime of Algorithm 1 to $O(T|G_T|^2)$, where $|G_T|$ is the number of selected Nyström points until the final timestep. A discussion on setting the hyperparameters η and b, controlling the tradeoff between regret and computation, is given in Appendix A.5.1.

MVCalc This subroutine calculates the posterior mean and variance for $\Delta(\cdot)$ using Gaussian process posterior calculations after projecting on the Nyström points G_t. We define the intermediate quantity

$$\Lambda_t = \left(K_{G_tS_t}K_{S_tG_t} + \lambda K_{G_tG_t}\right)^{-1},$$

which is useful in these updates. Note that the algorithms store and track the local variables $K_{GG}^{-1}, \Lambda_t, \tilde{\mathbf{y}}_t$ across time steps. It needs to incrementally invert K_{GG}^{-1} and Λ_t as the time steps continue. For Λ_t, if G_t does not change, it does this using the Sherman-Morrison formula:

$$\Lambda_t = \Lambda_{t-1} - \frac{\Lambda_{t-1}\hbar_{G_t}(x_t)\hbar_{G_t}(x_t)^\top\Lambda_{t-1}}{1+\hbar_{G_t}(x_t)^\top\Lambda_{t-1}\hbar_{G_t}(x_t)}. \tag{2.11}$$

This update takes $|G_t|^2$ time. In the case that G_t changes, let $a = K_{G_t}(x_t)$ and $c = \hbar(x_t, x_t)$, and set Λ_{t+1} as follows:

$$\Lambda_{t+1} = \begin{bmatrix} K_{G_tS}K_{SG_t} + aa^\top & K_{G_tS}^\top a + ca \\ a^\top K_{SG_t} + ca & a^\top a + c^2 \end{bmatrix}^{-1}. \tag{2.12}$$

We can use the Schur complement block-matrix inverse identity to evaluate the above which takes $O(t|G_t|)$ time. Similarly, for $K_{G_tG_t}^{-1}$, we write it in block-matrix form as:

$$K_{G_tG_t}^{-1} = \begin{bmatrix} K_{G_{t-1}G_{t-1}} & K_{G_{t-1}}(x_t) \\ K_{G_{t-1}}(x_t)^\top & \hbar(x_t, x_t) \end{bmatrix}^{-1}, \tag{2.13}$$

computable using Schur complements in $|G_t|^2$ time.

2.5.2 Theoretical guarantee

For a given set $X_T = \{x_1, \ldots, x_T\}$, all our bounds are stated in terms of the effective dimension of the matrix $K_T = K_{X_T X_T}$, as described in [48].

Definition 2.6. *The effective dimension of K_T with regularization $\lambda > 0$ is defined as $d_{eff}(\lambda, T) = \mathrm{Tr}(K_T(K_T + \lambda I)^{-1})$.*

Intuitively, the effective dimension is a measure of the number of dimensions in the feature space that are needed to capture data variations. Having a smaller effective dimension enables learning the unknown h_q with fewer samples. If the empirical kernel matrix K_T has eigenvalues $(\lambda_1, \ldots, \lambda_T)$, the effective dimension can equivalently be written as $d_{\text{eff}}(\lambda, T) = \sum_{t=1}^{T} \frac{\lambda_t}{\lambda_t + \lambda}$. Thus, if the eigenvalues decay quickly, the denominator will dominate for most summands and d_{eff} will be small. If the kernel is finite dimensional, with dimension s, then only the first s terms in the summation will be nonzero and $d_{\text{eff}} \leq s$. Having a small effective dimension makes the problem of learning the unknown objective function easier and hence improves our regret guarantee. This quantity is inspired by classical work in statistics [48, 114].

It is instructive to bound d_{eff} for different kernels. We relate d_{eff} to the information gain $\tilde{\gamma}(\lambda, T)$ [96], noting that

$$d_{\text{eff}}(\lambda, T) = \sum_{t=1}^{T} \frac{\frac{\lambda_t}{\lambda}}{\frac{\lambda_t}{\lambda} + 1} \leq \sum_{t=1}^{T} \log\left(1 + \frac{\lambda_t}{\lambda}\right) = \tilde{\gamma}(\lambda, T),$$

where we used the inequality $\frac{x}{x+1} \leq \log(1+x)$ for $x \geq -1$. Srinivas et al. [96] provides bounds on $\tilde{\gamma}(\lambda, T)$ for various kernels. For the (most popular) Gaussian kernel with dimension d, they show that $\tilde{\gamma}(\lambda, T) \leq \log(T)^{d+1}$ which holds under the assumption that the eigenvalues λ_t are square summable. We plot the exact $d_{\text{eff}}(\lambda, T)$ for the Gaussian kernel in Figure A.2 thereby verifying this bound empirically. For the linear kernel with dimension d, we have $\tilde{\gamma}(\lambda, T) \leq d \log(T)$. In our experimental results (see Section A.6), we use a composite over three constituent kernels and Theorems 2 and 3 of Krause and Ong [63] bound d_{eff} for the product or sums of kernels, when $\tilde{\gamma}(\lambda, T)$ is bounded for each constituent. This all ensures our regret bounds are sublinear in practice.

Now, we are ready to state our main result.

Theorem 2.4. *Let Assumption 2.1 hold and assume that ϵ_t are i.i.d. centered sub-Gaussian (i.e., light tailed) noise. Then MNN-UCB (Algorithm 1) obtains the following regret:*

(a) When all h_q are BP functions, we have that $\mathbb{E}[\mathcal{R}_{BP}(T)] \leq O\left(\sqrt{T}\left(B\sqrt{\lambda d_{\mathit{eff}}} + d_{\mathit{eff}}\right)\right)$

(b) When all h_q are WS functions, we have that $\mathbb{E}[\mathcal{R}_{WS}(T)] \leq O\left(\sqrt{T}\left(B\sqrt{\lambda d_{\mathit{eff}}} + d_{\mathit{eff}}\right)\right)$

We see that Lemma 2.1 and our algorithm design enables us to relate our notion of regret with the pointwise notion of regret from Zenati et al. [113].

Proof of Theorem 2.4(a). We define $R_t = \sum_{j=1}^{t} r_j$ where $r_t = \sup_{v \in V} h_{u_t}(v|S_{t_{u_t},u_t}) - h_{u_t}(v_t|S_{t_{u_t},u_t})$. Notice R_t is different from $\mathcal{R}_{\mathrm{BP}}(t)$. From Lemma 2.1 for all q, and with $\mathcal{R}_{\mathrm{BP}}(T)$ defined in Equation (2.8), we have $\mathcal{R}_{\mathrm{BP}}(T) \leq \sum_{t=1}^{T} r_t = R_T$. We model the problem as a contextual bandit problem in the vein of [113]. Here, the context in round t is $z_t = (\phi_{u_t}, S_{t_{u_t},u_t})$. We next invoke Theorem 4.1 in [113] to complete our result. □

Proof of Theorem 2.4(b). Define r_t and R_t as in the proof of Theorem 2.4(a). From Lemma 2.2 applied to each h_q, it follows that

$$\mathcal{R}_{\mathrm{WS}}(T) \leq \sum_{k=1}^{m} \sum_{t=1}^{T} \mathbb{I}(u_t = k) r_t = R_T.$$

As in the proof of Theorem 2.4(a), combining Theorem 4.1 of [113] with the above inequality, our argument is complete. □

2.6 No-Regret Separate Feedback

Algorithm 4 MNN-UCB-Separate (modified Algorithm 1)

Line 14 of Algorithm 1 replaced with the following:

1: Calculate distortion $D_t \leftarrow (1 - \frac{1}{T_{u_t}})^{T_{u_t} - |S_{u_t}| - 1}$.
2: Obtain submodular feedback $y_{f,t} = f_{u_t}(v_t|S_{u_t}) + \epsilon_{f,t}/2$ and $y_{g,t} = g_{u_t}(v_t|S_{u_t}) + \epsilon_{g,t}/2$.
3: Apply distortion to obtain overall feedback $y_t = D_t y_{f,t} + y_{g,t} + (1 - D_t) l_{u_t,1}(v_t)$.

In the previous section, we obtained sublinear α-regret with respect to the offline greedy baseline. Here, we show we can do the same relative to the "distorted" greedy baseline (Section 2.4.2). That is, our selection-rule should be approximately greedy w.r.t. the distorted $\pi(\cdot)$ function in Section 2.4. This is possible under the stronger separate feedback model in Section 2.3. As in Section 2.4, we

define for each function q the modular lower bound $l_{q,1}(S)$, the totally normalized submodular function $f_{q,1}$ and also $\pi_{j,\phi_q}(v|A)$ defined as:

$$\left(1 - \frac{1}{T_q}\right)^{T_q - j - 1} f_{q,1}(v|A) + g_q(v|A) + l_{q,1}(v). \tag{2.14}$$

The stronger notion of regret for BP functions with respect to the weighted greedy baseline, $\mathcal{R}_{\text{BP},2}(T)$ is defined as:

$$\sum_{q=1}^{m} \min\left\{1 - \frac{\kappa_{f,q}}{e},\ 1 - \kappa_q^g\right\} h_q(S_q^*) - h_q(S_q). \tag{2.15}$$

The algorithm that obtains sublinear regret with respect to this stronger baseline utilizes these π_{j,ϕ_q} functions. Also, we use $\pi_{j,q} = \pi_{j,\phi_q}$ interchangeably for readability.

2.6.1 MNN-UCB-SEPARATE Algorithm.

Algorithm 4 is quite similar to Algorithm 1 — line 14 of Algorithm 1 is modified to the three steps of Algorithm 4. That is, the feedback, now obtained separately as $y_{f,t}$ for the submodular and as $y_{g,t}$ for the supermodular part, is aggregated in line 3 of Algorithm 4 as per π_{j,ϕ_q}. To evaluate these expressions, the optimizer also requires the following:

Assumption 2.2. *(a) The modular lower bound $l_{q,1}(\cdot)$ is known by the optimizer. (b) The optimizer has access to two oracles that provide it with separate feedback for the submodular $f_q(\cdot|\cdot)$ and supermodular $g_q(\cdot|\cdot)$ part. (c) The number of items for each user T_q is known by the optimizer.*

For Assumption (2a), $l_{q,1}(\cdot)$ is precisely specified using $|V|$ entries while the submodular function requires $2^{|V|}$ entries to be specified. Hence, the submodular f_q is still mostly unknown as knowing $l_{q,1}$ is a weaker assumption than the offline setting. We additionally provide an alternate (slightly weaker) result in Appendix A.5.6 without this assumption. For Assumption (2b), this can be estimated in applications using surveys with multiple questions rather than a single rating. For Assumption (2c), if T_q is not known beforehand, "guess and double" techniques (Appendix A.5.7) can be used, the effects of which result in a bounded additive term.

2.6.2 Performance with Separate Feedback

The guarantee for Algorithm 4 is the following:

Theorem 2.5. *Let Assumptions 2.1 and 2.2 hold and assume that ϵ_t are i.i.d centered sub-Gaussian noise. Then, when all h_q are BP functions, Algorithm 4 yields*

$$\mathbb{E}[\mathcal{R}_{BP,2}(T)] \leq O\left(\sqrt{T}\left(B\sqrt{\lambda d_{\mathit{eff}}} + d_{\mathit{eff}}\right)\right).$$

The proof follows similar lines as Theorem 2.4, with the key difference being that instantaneous regret is now defined in terms of the distorted objective.

Proof of Theorem 2.5. First, define the following notation:

$$\begin{aligned}
l_{q,1}(S) &= \sum_{j \in S} f_q(j|V\backslash\{j\}),\\
f_{q,1}(S) &= f_q(S) - l_{q,1}(S),\\
l_{q,2}(S) &= \sum_{j \in S} g_q(j|\emptyset),\\
g_{q,1}(S) &= g_q(S) - l_{q,2}(S),\\
l_q(S) &= l_{q,1}(S) + l_{q,2}(S).
\end{aligned}$$

We restrict attention to the q-th function h_q. Recall that $S_{j,q}$ refers to the first j elements chosen for h_q.

Let the distorted objective for user q when selecting the j-th item in the set be:

$$\pi_{j,q}(S) = \left(1 - \frac{1}{T_q}\right)^{T_q - j} f_{q,1}(S) + g_{q,1}(S) + l_q(S).$$

Additionally, define $\Lambda_{j,q}$ as follows:

$$\Lambda_{j,q}(x, A) = \left(1 - \frac{1}{T_q}\right)^{T_q - (j+1)} f_{q,1}(x|A) + g_{q,1}(x|A) + l_q(x).$$

As previously, we define the instantaneous regret at round t as the difference between the maximum possible utility that is achievable in the round and the actual received utility. However,

this time, r_t is defined in terms of the distorted objective. *This is a key difference from the earlier arguments that is crucial to the current proof.* Define the accumulated instantaneous regret until round t as

$$R_t = \sum_{j=1}^{t} r_j,$$

where

$$r_t = \sup_{v \in V} \Lambda_{q,t_{u_t}}(v, S_{u_t,t-1}) - \Lambda_{q,t_{u_t}}(v_t, S_{u_t,t-1}).$$

Recognize that R_t is different than $\mathcal{R}_t$. From Lemma 2.3 applied to each h_q, it follows that

$$\mathcal{R}_T \leq \sum_{q=1}^{m} \sum_{t=1}^{T} \mathbb{I}(u_t = q) r_t = R_T. \tag{2.16}$$

Now, we can model the problem of the present work as a contextual bandit problem in the vein of [113]. Here, the context in the t-th round is $z_t = (\phi_{u_t}, S_{t_{u_t},u_t})$. Now we invoke Theorem 4.1 in [113], Thus, we have that

$$\mathbb{E}[R_T] \leq O\left(\sqrt{T}\left(B\sqrt{\lambda d_{\text{eff}}} + d_{\text{eff}}\right)\right).$$

Combining this with Inequality (2.16), our argument is complete.

□

For some applications, there may not be enough information for Assumption (2a). In Appendix A.5.6, we provide an alternative argument without this assumption where the α is slightly reduced to $\min\left\{1 - \frac{1}{e},\ 1 - \kappa_q^g\right\}$. However, the heat map in Appendix A.5 illustrates the bounds are still better than the vanilla greedy α from [5] for most choices of $\kappa_{f,q}, \kappa_q^g$.

2.7 Numerical Experiments

From MovieLens [45], we obtain a ratings matrix $M \in \mathbb{R}^{900\times1600}$, where $M_{t,j}$ is the rating of the t^{th} user for the j^{th} movie. Using this dataset, we instantiate an interactive BP maximization problem, as formulated in Vignette 2.1. We cluster the users into $m = 10$ groups using the k-means algorithm and design a BP objective for each user-group. The objective for the q_{th} group is decomposed as $h_q(A) = \sum_{v\in A} m_q(v) + \lambda_1 f_q(A) + \lambda_2 g_q(A)$, where the modular part $m_q(v)$ is the average rating

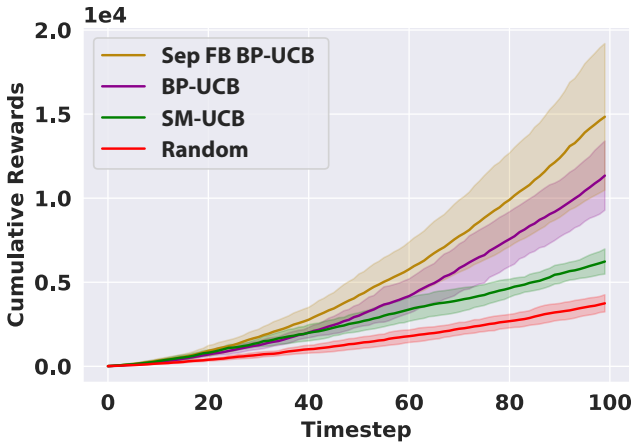


Figure 2.1: Algorithm 1 (magenta, green) and Algorithm 4 (gold) applied to the MovieLens dataset. The highlighted region shows the standard deviation over 10 random trials.

for movie v amongst all users in group k. The concave-over-modular submodular part encourages the recommender to maintain a balance across genres in chosen suggestions. By contrast, the supermodular function is designed to encourage the optimizer to exploit complementarities within genres. The constants λ_1, λ_2 are chosen so that the supermodular part slightly dominates the submodular part, since previous work already studies primarily submodular functions. In Figure 2.1, Algorithm 4 (gold) performs slightly better than Algorithm 1 (magenta), as expected from our results. If we provide MNN-UCB submodular feedback i.e $\sum m_q(v) + f_q(A)$ instead of the entire h_q, then it performs notably worse (green, labeled SM-UCB). This underscores the pitfalls of viewing a non-submodular problem as purely submodular.

Active learning experiments are given in Appendix A.2.1.

Chapter 3

Choice-Driven Learning and Peer Model Probing

Traditional supervised learning theory typically assumes a single learner observing data drawn from a fixed distribution. However, this assumption is increasingly violated in modern machine learning markets, such as recommendation platforms and large language model (LLM) services. In these ecosystems, multiple learners operate on the same pool of users, and data is not assigned randomly. Instead, users choose which platform to engage with based on how well that platform serves their specific needs or preferences. Consequently, the data distribution observed by a learner is a function of the learner's own performance and the choices available in the market. This setting is increasingly garnering interest in the machine learning community [13, 31, 36, 94, 98].

This coupling between model performance and user selection creates a feedback loop. As a learner optimizes for its current user base, it becomes increasingly specialized to that subpopulation. While this minimizes "local" loss on observed users, it often degrades performance on the unobserved population, a phenomenon termed *overspecialization*. Once a learner is overspecialized, it gets caught in an informational trap: it cannot learn to serve new users because it never observes them, and it never observes them because it cannot serve them. At a societal level, this dynamic fuels the formation of algorithmic echo chambers [6, 27, 53, 60], where platforms fragment the population rather than learning a robust, globally capable model.

Independently, another trend has become relevant in modern machine learning systems that

has implications for the overspecialization problem: techniques such as knowledge distillation and training on synthetic data are becoming ubiquitous, particularly in the training of Large Language Models [49, 107]. While these methods are typically employed to improve reasoning capabilities or computational efficiency (through compression of data), they introduce a structural change to the learning dynamic. Models are no longer limited to learning from organic user data, but can also "probe" other models to acquire synthetic labels. This enables learners to observe signals outside their siloed data distributions. This chapter studies whether probing mechanisms in machine learning markets can mitigate overspecialization.

This chapter models a market where users select learners based on a combination of inherent preference and predictive loss, and analyzes the resulting dynamics through a game-theoretic lens. The main findings are: (1) standard Multi-learner Streaming Gradient Descent (MSGD) can converge to overspecialized equilibria with arbitrarily poor global performance; (2) a new algorithm, MSGD with Probing (MSGD-P), where learners mix organic gradient updates with pseudo-labeled queries to peer models, provably converges to stationary points of a modified potential; (3) the stationary points of MSGD-P yield bounded full-population loss under identifiable informational conditions; and (4) experiments on MovieLens, US Census, and Amazon Sentiment datasets validate these findings.

The remainder of the chapter is organized as follows. Section 3.2 formalizes the problem setting, including user preferences, platform choice, and the learning objective. Section 3.3 analyzes the failure of standard MSGD dynamics and characterizes the overspecialization trap. Section 3.4 introduces peer probing, proves convergence of MSGD-P, and establishes performance guarantees. Section 3.5 presents numerical experiments.

3.1 Related Work

Our work sits at the intersection of three lines of research. We study a *multi-learner* setting where users select among competing platforms based on a combination of *inherent preferences and predictive quality*: a model richer than pure loss-minimization or uniform-random selection. Motivated by modern distillation practices, we analyze *peer-model probing* as a mechanism to mitigate overspecialization.

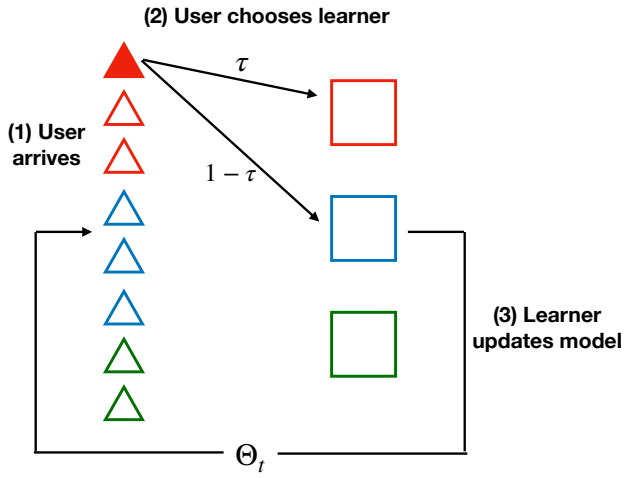


Figure 3.1: Illustration of the online multi-learner problem setting. The borders of users represent their highest ranked learner $\pi(z)$. For further details, see Section 3.2.

Performative Prediction and Endogenous Distribution Shift. Our setting is an instance of performative prediction [44, 73, 82], where the deployment of a model influences the distribution of data it subsequently observes. The multi-learner extensions of this framework [37, 66, 75, 83, 104, 105, 117] provide general tools for analyzing endogenous distribution shift, including settings with explicitly strategic feature manipulation. We specialize to the user-choice setting, where distribution shift arises purely from selection rather than manipulation. This focus enables us to precisely characterize overspecialization and to study peer-model probing as a mitigation strategy.

Learning under User Agency. A body of work studies learning dynamics when users are treated as independent agents rather than passive data points. In single-learner settings, Hashimoto et al. [47] show that empirical risk minimization can cause minority groups to opt out, creating a feedback loop that further degrades minority performance. Zhang et al. [115] study a related dynamic where underrepresented groups receive worse predictions, leading to further data scarcity. James et al. [58] analyze participatory data collection where users choose whether to contribute data. Ben-Porat and Tennenholtz [7] study best-response dynamics in strategic classification. Cherapanamjeri et al. [23] and Harris et al. [46] study settings where data quality or availability depends on the learner's past performance. We build on this perspective in the multi-learner setting, where inter-learner interactions create additional feedback dynamics.

Several works study multi-learner user-choice settings with explicitly strategic users who optimize

their own utility functions [7, 8, 56, 57, 94]. Ginart et al. [36] provide both empirical and theoretical analysis of how competition drives specialization: their Theorems 4.1–4.3 establish risk ratio bounds showing that competing predictors perform worse on the general population than a single predictor would. Their analysis considers batch retraining dynamics and characterizes the *existence* of performance gaps due to competition. We complement this by analyzing streaming gradient-based dynamics, proving that such dynamics *converge to* overspecialized equilibria (Theorem 3.3), and proposing probing as a mitigation. Kwon et al. [65] study a related setting where learners may purchase user data, providing primarily empirical analysis of the resulting market dynamics.

Most closely related to our work, Dean et al. [31] and Su and Dean [98] analyze gradient-based dynamics in choice-driven settings. Su and Dean [98] introduce the MSGD algorithm and prove convergence to stationary points of an aggregate loss across learners. We build directly on their framework: our Algorithm 5 adapts MSGD to our user selection rule (Definition 3.1), and our potential function (Equation 3.1) extends theirs. Our Theorem 3.2 reproves their convergence result using stochastic approximation techniques, which we believe better illuminates the underlying dynamics. Beyond convergence, we analyze generalization to users outside each learner's observed population—a consideration absent from prior work—and introduce peer probing as a mechanism to restore global competence. Bose et al. [13] propose intelligent initialization schemes to improve outcomes in similar settings, but also focus on losses on the observed distribution.

Knowledge Distillation and Peer Learning. Our probing mechanism draws inspiration from knowledge distillation [49] and self-training methods [90, 112], where a learner augments its training set using pseudo-labels from another model. These techniques are ubiquitous in modern practice, particularly for compressing large language models or generating synthetic reasoning traces [107, 109, 110]. A natural question is how our work relates to online/mutual distillation; unlike Deep Mutual Learning [116] and codistillation [3], which assume shared or randomly partitioned data, our setting couples learning with user-driven selection, yielding endogenous heterogeneity and novel multi-agent dynamics absent from prior distillation analyses.

3.2 Problem Setting

Users and Learners. Consider a setting with m service providers (learners) serving a population of users distributed according to $\mathcal{P}$ over $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$, where $\mathcal{X} \subseteq \mathbb{R}^d$ denotes covariates and $\mathcal{Y}$ denotes labels ($\mathcal{Y} \subseteq \mathbb{R}$ for regression, $\mathcal{Y} \subseteq \{1, \ldots, C\}$ for classification). We write $\mathcal{P}_X$ for the marginal distribution over covariates and $\mathcal{P}_{Y|X}(\cdot \mid x)$ for the conditional label distribution. When densities exist, we denote them by p_X and $p_{Y|X}$.

Each learner $i \in [m] := \{1, \ldots, m\}$ maintains a model with parameters θ_i, and we write $\Theta = (\theta_1, \ldots, \theta_m)$ for the joint parameter vector. The loss $\ell(x, y; \theta)$ measures the cost incurred by a model with parameters θ on a user with features x and label y. For regression, we consider linear predictors $h_\theta(x) = x^\top \theta$ with squared loss $\ell_{\mathrm{SQ}}(x, y; \theta) = (y - x^\top \theta)^2$. For classification with C classes, we use cross-entropy loss $\ell_{\mathrm{CE}}(x, y; \theta) = -\sum_{c=1}^{C} y_c \log h_\theta(x)_c$, where $h_\theta(x)_c = e^{x^\top \theta_c} / \sum_{j=1}^{C} e^{x^\top \theta_j}$ is the softmax output.

Assumption 3.1. *The distribution $\mathcal{P}$ has continuous density $p_Z(z) = p_X(x) p_{Y|X}(y|x)$ with p_X supported on $\{x : \|x\| \le R\}$ for some $R > 0$. For regression, $p_{Y|X}$ is supported on $[-Y_{\max}, Y_{\max}]$.*

Under this assumption, both loss functions satisfy standard regularity conditions; see Lemma B.5 in Appendix B.1.

User Preferences and Platform Choice. A central feature of our framework is that users have *inherent preferences* over platforms that exist independently of current model quality. These preferences capture factors such as brand loyalty, familiarity, network effects, or historical habits. We encode them via a function $\pi : \mathcal{Z} \to [m]$, where $\pi(z)$ denotes the platform that user z intrinsically prefers. This function is exogenous and fixed throughout learning.

The preference function π induces a partition of the user space. Let $S_i = \{z \in \mathcal{Z} : \pi(z) = i\}$ denote users who prefer platform i, with $\mathcal{P}_i = \mathcal{P}|_{S_i}$ the corresponding conditional distribution and $\alpha_i = \Pr_{z \sim \mathcal{P}}[\pi(z) = i]$ the fraction of users preferring platform i. Users do not only follow their inherent preferences; they also consider predictive quality. We model this tradeoff as follows:

Definition 3.1 (User Selection Rule)**.** *Given models Θ, user z selects platform*

$$M(z;\Theta) = \begin{cases} \pi(z) & \text{with probability } \tau, \\ \arg\min_{i\in[m]} \ell(z;\theta_i) & \text{with probability } 1-\tau. \end{cases}$$

The parameter $\tau \in [0,1]$ governs how strongly inherent preferences influence user behavior: $\tau = 1$ means users follow intrinsic preferences entirely, while $\tau = 0$ means they select the loss-minimizing platform. This generalizes Su and Dean [98], who assume users either minimize loss or choose uniformly at random.

Model quality also induces a partition of users. Let $Z_i(\Theta) = \{z : i = \arg\min_{j\in[m]} \ell(z;\theta_j)\}$ denote users for whom platform i achieves minimal loss, with $\mathcal{D}_i(\Theta) = \mathcal{P}|_{Z_i(\Theta)}$ the conditional distribution and $a_i(\Theta) = \Pr_{z\sim\mathcal{P}}[z \in Z_i(\Theta)]$ the population fraction. Under Definition 3.1, learner i observes users from the mixture

$$\mathcal{O}_i(\Theta) = \tau\alpha_i\mathcal{P}_i + (1-\tau)a_i(\Theta)\mathcal{D}_i(\Theta),$$

where $\mathcal{O}_i(\Theta)$ is a sub-probability measure with total mass $w_i(\Theta) = \tau\alpha_i + (1-\tau)a_i(\Theta)$.

Dynamics and Learning Objective. We consider an online setting where learners interact with users over T timesteps, illustrated in Figure 3.1. At each step t:

1. A user $z^t \sim \mathcal{P}$ arrives and selects platform $M(z^t;\Theta^t)$.
2. The selected learner observes z^t, incurs loss $\ell(z^t;\theta^t_{M(z^t;\Theta^t)})$, and updates its parameters.

Learning Objective. Each learner's goal is to minimize the *full-population risk*

$$\mathcal{R}(\theta) = \mathbb{E}_{z\sim\mathcal{P}}[\ell(z;\theta)].$$

We write $\theta^\star = \arg\min_\theta \mathcal{R}(\theta)$ for the population-optimal model and $\epsilon = \mathcal{R}(\theta^\star)$ for the Bayes risk. This objective differs from prior work [31, 98], which focuses on the "local" loss over each learner's observed distribution $\mathcal{O}_i(\Theta)$. We focus on full-population risk because the signature of overspecialization is precisely the gap between local and global performance: a learner may achieve low loss on $\mathcal{O}_i(\Theta)$ while performing poorly on users outside its observed base. Understanding when this gap emerges and how to prevent it is the central question of this chapter.

Algorithm 5 Multi-learner Streaming Gradient Descent (MSGD) [98]

Require: Loss function $\ell(\cdot,\cdot) \geq 0$; initial models $\Theta^0 = (\theta_1^0, \ldots, \theta_m^0)$; learning rate $\{\eta^t\}_{t\geq 1}$

1: **for** $t = 0, 1, 2, \ldots, T$ **do**
2: Sample user $z^t \sim \mathscr{P}$
3: User selects learner $i = M(z^t; \Theta^t)$
4: $\theta_i^{t+1} \leftarrow \theta_i^t - \eta^t \nabla_\theta \ell(z^t; \theta_i^t)$
5: **end for**
6: **return** Θ^T

3.3 The Failure of Standard Learning Dynamics

We now analyze Multi-learner Streaming Gradient Descent (MSGD), the standard algorithm for this setting introduced by Su and Dean [98], Algorithm 5.

3.3.1 Algorithm and Assumptions

In order to study the convergence behavior of the algorithm, we make the following standard assumptions on learning rates and loss geometry, which are the same as in Su and Dean [98].

Assumption 3.2. *The learning rates satisfy* $\sum_{t=1}^{\infty} \eta^t = \infty$ *and* $\sum_{t=1}^{\infty} (\eta^t)^2 < \infty$.

Assumption 3.3. *For any* $\theta \neq \theta'$, *there exists* $d_0 > 0$ *such that for all* $d < d_0$, *the set* $\{z : |\ell(z;\theta) - \ell(z;\theta')| < d\}$ *has Lebesgue measure at most* d.

3.3.2 Convergence to Stationary Points

In MSGD, each learner i optimizes its expected loss over observed users. Define the potential function $f(\Theta)$ as the sum of the expected losses of all learners on the distributions that they observe.

$$f(\Theta) = \sum_{i=1}^{m} \mathbb{E}_{\mathcal{O}_i(\Theta)}[\ell(z;\theta_i)], \tag{3.1}$$

Note that this is the sum of the "local" losses of each of the learners on their observed distribution, unlike the global risk $\mathcal{R}(\cdot)$ that was introduced above.

In order to show convergence, we make the same boundedness assumptions as in Su and Dean [98].

Assumption 3.4. *The parameter sequence* $\{\Theta^t\}_{t\geq 0}$ *is almost surely bounded:* $\sup_{t\geq 0} \|\Theta^t\| < \infty$. *Moreover, the set* $\{\Theta : \nabla f(\Theta) = 0\}$ *is compact.*

Our approach uses stochastic approximation [12] and differs from that of Su and Dean [98]. The key insight is that f serves as a Lyapunov function: despite each learner optimizing over a different, endogenously-determined user distribution, the aggregate of local losses forms a coherent potential. This is surprising because multi-agent gradient dynamics often cycle or diverge [72]; The following lemma, proved using standard stochastic approximation arguments, establishes this connection.

Lemma 3.1. *Let Assumptions 3.1–3.4 hold. Then the iterates $\{\Theta^t\}$ of Algorithm 5 converge to a compact connected internally chain transitive invariant set of the ODE $\dot{\Theta} = -\nabla f(\Theta)$.*

See Appendix B.1.1 for definitions of the dynamical-systems terms used in Lemma 3.1. The lemma shows that the discrete stochastic dynamics behave, in the limit, like the continuous gradient flow on f. Since f decreases along trajectories of this ODE, i.e., $\frac{d}{dt} f(\Theta(t)) = -\|\nabla f\|^2 \leq 0$, the only invariant sets are stationary points. This yields our main convergence result.

Theorem 3.2. *Let Assumptions 3.1–3.4 hold. Then the iterates $\{\Theta^t\}$ of Algorithm 5 converge to the set of stationary points $\{\Theta : \nabla f(\Theta) = 0\}$ almost surely.*

The formal proof follows as a special case of the MSGD-P convergence analysis (Theorem 3.6), whose proof appears in Section 3.4.

3.3.3 The Overspecialization Trap

While Theorem 3.2 guarantees convergence, the stationary points may be highly undesirable. A learner cannot improve on users it never observes, and it never observes users it cannot serve well. This feedback loop is the overspecialization trap.

The following theorem shows that this trap can be severe: MSGD can converge to equilibria where some learners have arbitrarily poor global performance, even when models with low full-population loss exist.

Theorem 3.3. *Let Assumptions 3.1–3.4 hold. For any $\tau \geq \frac{1}{2}$ and any choice of ϵ, Γ with $0 < \epsilon < \Gamma$, there exists an instance $\mathcal{G} = (\mathcal{P}, \ell, \pi, \tau)$ such that:*

1. *There exists $\theta^\star$ with $\mathcal{R}(\theta^\star) \leq \epsilon$.*
2. *The MSGD iterates converge to a unique stationary point $\bar{\Theta}$ where $\mathcal{R}(\bar{\theta}_i) \geq \Gamma$ for some learner $i \in [m]$.*

The key mechanism is as follows. When $\tau \geq \frac{1}{2}$, inherent preferences dominate at equilibrium: the loss-induced partition collapses to the ranking partition, so that $Z_i(\Theta) = S_i$ for all i. Each learner optimizes exclusively for users who intrinsically prefer it, arriving at

$$\bar{\theta}_i = \arg\min_{\theta} \mathbb{E}_{z \sim \mathscr{P}_i}[\ell(z; \theta)].$$

This specialization occurs regardless of whether a better global model exists. In the constructed instance below, learner 1 achieves *zero* loss on its observed population $\mathscr{P}_1$ while its full-population loss exceeds Γ. The learner has perfectly fit its niche while becoming arbitrarily poor globally. This dynamic formalizes the echo chamber phenomenon that platforms become increasingly specialized to their existing audience, unable to learn models that serve the broader population.

We now make this precise via a concrete construction.

Example 3.1. *Specify the family of instances $\mathcal{G}_{bad}(\tau, C)$ as follows.*

1. *Distribution $\mathscr{P}$: is defined as a mixture of subpopulations. $\mathscr{P} = \alpha\mathscr{P}_1 + (1-\alpha)\mathscr{P}_2$. Here, $\{\mathscr{P}_1, \mathscr{P}_2\}$ are 2 subpopulations. For either subpopulation, the covariates are generated from the zero mean and unit-variance uniform distribution:*

$$x \sim \mathit{Unif}([-\sqrt{3}, \sqrt{3}]) \quad \mathit{for} \quad (x, y) \sim \mathscr{P}_i, \tag{3.2}$$

   For each subpopulation, the response variable is generated as:

$$y = Cx_1 \quad \mathit{for} \quad (x, y) \sim \mathscr{P}_1 \tag{3.3}$$

$$y = -x_1 \quad \mathit{for} \quad (x, y) \sim \mathscr{P}_2 \tag{3.4}$$

2. *Loss function: $\ell(x, y, \theta) = (y - \theta^T x)^2$ is the squared loss*

3. *Ranking $\pi(z) = i$ for $(x, y) \sim \mathscr{P}_i$.*

The next lemma quantifies the risk gap between specialists and the global optimum in this family.

Lemma 3.4. *Consider the 1-D bad-outcome family in Example 3.1 with mixture weight $\alpha \in (0, 1)$ and slope parameter $C > 1$, and let $\mathcal{R}(\theta) = \mathbb{E}_{(x,y)\sim\mathscr{P}}[(y - \theta x)^2]$.*

(i) The least-squares predictor on the full mixture, $\theta^\star = \alpha C - (1-\alpha)$, satisfies

$$\mathcal{R}(\theta^\star) = \alpha(1-\alpha)(C+1)^2.$$

(ii) The specialist trained on $\mathcal{P}_1$ is $\bar{\theta}_1 = C$, and its mixture risk is

$$\mathcal{R}(\bar{\theta}_1) \;=\; (1-\alpha)(C+1)^2.$$

(iii) The specialist trained on $\mathcal{P}_2$ is $\bar{\theta}_2 = -1$, and its mixture risk is

$$\mathcal{R}(\bar{\theta}_2) \;=\; \alpha\,(C+1)^2.$$

Proof. We have that $\mathbb{E}[x] = 0$ and $\mathbb{E}[x^2] = 1$, so that the squared risk reduces to $(\beta - \theta)^2$ when the true slope is β.

(i) Global compromise. On the mixture, the conditional label is linear: $y = \beta x$ with $\beta = \alpha C + (1-\alpha)(-1) = \alpha C - (1-\alpha)$. For squared loss with $\mathbb{E}[x] = 0$ and $\mathbb{E}[x^2] = 1$, the population least-squares minimizer is the regression slope, so $\theta^\star = \beta$. The corresponding risk is

$$\mathcal{R}(\theta^\star) = \alpha\,(C-\beta)^2 + (1-\alpha)\,(-1-\beta)^2 = \alpha(1-\alpha)(C+1)^2.$$

(ii) Specialist for $\mathcal{P}_1$. Training least squares on $\mathcal{P}_1$ alone yields $\bar{\theta}_1 = C$ since $\mathbb{E}[x\,y] = C$ and $\mathbb{E}[x^2] = 1$. The mixture risk is

$$\mathcal{R}(\bar{\theta}_1) = \alpha\,(C-C)^2 + (1-\alpha)\,(-1-C)^2 = (1-\alpha)(C+1)^2.$$

(iii) Specialist for $\mathcal{P}_2$. Training least squares on $\mathcal{P}_2$ gives $\bar{\theta}_2 = -1$ (its true slope). The mixture risk is

$$\mathcal{R}(\bar{\theta}_2) = \alpha\,(C+1)^2 + (1-\alpha)\,(-1-(-1))^2 = \alpha\,(C+1)^2.$$

□

It remains to show that MSGD must converge to these specialists. The following lemma establishes that when $\tau \geq \frac{1}{2}$, the only stationary points exhibit full specialization.

Lemma 3.5. *Let $(\tilde{\theta}_1, \tilde{\theta}_2) \in \{\nabla f(\Theta) = 0\}$ be a stationary point of $f(\Theta)$ (Definition 3.1). Then, under the assumption that $\tau \geq \frac{1}{2}$, it must be true that $\mathcal{D}_1(\tilde{\theta}_1, \tilde{\theta}_2) = \mathcal{P}_1$ and $\mathcal{D}_2(\tilde{\theta}_1, \tilde{\theta}_2) = \mathcal{P}_2$.*

Proof. The proof proceeds by considering the decisions of the loss-minimizing users, and shows that the stationary point condition implies the conditions on $\mathcal{D}_i(\Theta)$ in the lemma statement. Consider any (x, y) from $\mathcal{P}_1$. For any $\theta \in \mathbb{R}$, we have that

$$\ell(x, y, \theta) = (Cx - \theta x)^2 = (C - \theta)^2 x^2. \tag{3.5}$$

Hence,

$$\arg\min_{i \in \{1,2\}} \ell(x, y, \theta_i) = \arg\min_{i \in \{1,2\}} (C - \theta_i)^2 x^2 = \arg\min_{i \in \{1,2\}} |C - \theta_i|.$$

Thus every loss-minimizing user in $\mathcal{P}_1$ picks the learner whose parameter is closer to C:

$$M(z; \Theta) = \arg\min_{i \in \{1,2\}} |C - \theta_i|, \quad z \in \mathcal{P}_1.$$

Likewise, every loss-minimizing user in $\mathcal{P}_2$ picks the learner whose parameter is closer to -1:

$$M(z; \Theta) = \arg\min_{i \in \{1,2\}} |-1 - \theta_i|, \quad z \in \mathcal{P}_2.$$

Hence, within each subpopulation $\mathcal{P}_i$, all loss-minimizing users make the same choice. Consequently, for any Θ, the observed-data distributions $\mathcal{D}_i(\Theta)$ can only take one of the following four forms:

1. $\mathcal{D}_1(\Theta) = \mathcal{P}$ and $\mathcal{D}_2(\Theta)$ has zero mass under $\mathcal{P}$
2. $\mathcal{D}_1(\Theta)$ has zero mass under $\mathcal{P}$ and $\mathcal{D}_2(\Theta) = \mathcal{P}$
3. $\mathcal{D}_1(\Theta) = \mathcal{P}_2$ and $\mathcal{D}_2(\Theta) = \mathcal{P}_1$
4. $\mathcal{D}_1(\Theta) = \mathcal{P}_1$ and $\mathcal{D}_2(\Theta) = \mathcal{P}_2$

For any stationary point $(\tilde{\theta}_1, \tilde{\theta}_2)$, we consider each case separately and show that all cases other than the last one lead to a contradiction.

Case 1: In this case, the second learner observes only negative labels, and the first observes a mix of both positive and negative labels. Hence, it should be intuitively true that $\tilde{\theta}_1 > \tilde{\theta}_2 > -1$. This would lead to a contradiction since this would imply that users from $\mathscr{P}_2$ would strictly prefer the second learner over the first one. Now, we can verify this formally. Define the total probability masses seen by each learner

$$M_1 = \tau\,\alpha \;+\; (1-\tau)\,\alpha \;+\; (1-\tau)(1-\alpha) \;=\; \alpha \;+\; (1-\tau)(1-\alpha), \tag{3.6}$$

$$M_2 = \tau\,(1-\alpha). \tag{3.7}$$

The stationary conditions are given by

$$\tilde{\theta}_1 = \arg\min_{\theta_1} \tau\,\alpha\,\mathbb{E}_{P_1}\big[(y-\theta_1\,x)^2\big] + (1-\tau)\,\alpha\,\mathbb{E}_{P_1}\big[(y-\theta_1\,x)^2\big] + (1-\tau)(1-\alpha)\,\mathbb{E}_{P_2}\big[(y-\theta_1\,x)^2\big], \tag{3.8}$$

$$\tilde{\theta}_2 = \arg\min_{\theta_2} \tau\,(1-\alpha)\,\mathbb{E}_{P_2}\big[(y-\theta_2\,x)^2\big]. \tag{3.9}$$

We can solve these optimization problems in closed form:

$$\tilde{\theta}_1 = \frac{\alpha\,C \;-\; (1-\tau)(1-\alpha)}{M_1}, \tag{3.10}$$

$$\tilde{\theta}_2 = -1. \tag{3.11}$$

Now,

$$\tilde{\theta}_1 - \tilde{\theta}_2 = \tilde{\theta}_1 + 1 = \frac{\alpha\,(C+1)}{M_1} > 0,$$

which holds for every $\tau \in [0,1]$ since $\alpha > 0$ and $C > 0$. Since $\tilde{\theta}_2 = -1$ and $\tilde{\theta}_1 > -1$, $\mathscr{P}_2$ users would strictly prefer learner 2 over learner 1, leading to a contradiction.

Case 2: In this case, the first learner observes only positive labels, and the second observes a mix of both positive and negative labels. Hence, it is easy to verify using a similar procedure as the previous case that $C > \tilde{\theta}_1 > \tilde{\theta}_2$, which would lead to a contradiction since $\mathscr{P}_1$ users would strictly prefer learner 1.

Case 3: We follow a similar argument as in Case 1. Define the total probability masses seen by

each learner

$$M_1 = \tau\,\alpha \;+\; (1-\tau)(1-\alpha), \tag{3.12}$$

$$M_2 = \tau\,(1-\alpha) \;+\; (1-\tau)\,\alpha. \tag{3.13}$$

The stationary points in this case are given by

$$\tilde{\theta}_1 \;=\; \frac{\tau\,\alpha\,C \;-\; (1-\tau)(1-\alpha)}{\tau\,\alpha \;+\; (1-\tau)(1-\alpha)}, \qquad \tilde{\theta}_2 \;=\; \frac{(1-\tau)\,\alpha\,C \;-\; \tau(1-\alpha)}{(1-\tau)\,\alpha \;+\; \tau(1-\alpha)}.$$

It is easy to verify that both $\tilde{\theta}_1$ and $\tilde{\theta}_2$ are greater than -1. Hence, the distance from -1 is given by

$$d_1 = \left|\tilde{\theta}_1 - (-1)\right| = \frac{\tau\,\alpha\,(C+1)}{\tau\,\alpha \;+\; (1-\tau)(1-\alpha)},$$
$$d_2 = \left|\tilde{\theta}_2 - (-1)\right| = \frac{(1-\tau)\,\alpha\,(C+1)}{(1-\tau)\,\alpha \;+\; \tau(1-\alpha)}.$$

A straightforward comparison gives

$$d_1 > d_2 \iff \tau\,M_2 > (1-\tau)\,M_1 \iff (1-\alpha)\,(2\tau-1) > 0 \iff \tau > \tfrac{1}{2}.$$

Therefore, whenever $\tau > \frac{1}{2}$, the free users from $\mathscr{P}_2$ strictly prefer learner 2 over learner 1. This contradicts the Case 3 hypothesis that $\mathscr{D}_1(\Theta) = \mathscr{P}_2$. Hence Case 3 cannot be a stationary point when $\tau > 1/2$. Having ruled out Cases 1–3, only the fourth configuration remains, completing the proof. □

Combining these ingredients yields the theorem.

Proof of Theorem 3.3. The proof follows by considering Example 3.1.

Part (i). The proof follows from Lemma 3.4. Choosing $\alpha = \frac{\epsilon}{(C+1)^2}$ satisfies the condition.

Characterizing the stationary points. From Lemma 3.2, we have that the MSGD iterates converge almost surely to the set $\{\nabla f(\Theta) = 0\}$. By Lemma 3.5, we have that $(\tilde{\theta}_1, \tilde{\theta}_2) \in \{\nabla f(\Theta) = 0\}$

if and only if $\mathcal{D}_1(\tilde{\theta}_1, \tilde{\theta}_2) = \mathcal{P}_1$ and $\mathcal{D}_2(\tilde{\theta}_1, \tilde{\theta}_2) = \mathcal{P}_2$. Hence, any stationary point must satisfy

$$\tilde{\theta}_1 = \arg\min_{\theta \in \mathbb{R}} \alpha \, \mathbb{E}_{z \sim \mathcal{P}_1}[\ell(z, \theta)]$$

and

$$\tilde{\theta}_2 = \arg\min_{\theta \in \mathbb{R}} (1 - \alpha) \, \mathbb{E}_{z \sim \mathcal{P}_2}[\ell(z, \theta)]$$

Clearly, $(\bar{\theta}_1, \bar{\theta}_2)$ is the unique solution to these equations, and MSGD must converge to this point almost surely.

Characterizing the loss. From Lemma 3.4, at the stationary point we have $\bar{\theta}_1 = C$ and $\bar{\theta}_2 = -1$, so the mixture risks are

$$\mathcal{R}(\bar{\theta}_1) = (1 - \alpha)(C + 1)^2, \qquad \mathcal{R}(\bar{\theta}_2) = \alpha(C + 1)^2.$$

Moreover, the optimal global (compromise) risk satisfies $\mathcal{R}(\theta^*) \leq \epsilon$ by our choice in Part (i). To ensure learner 1's risk exceeds Γ, choose

$$C = \sqrt{\Gamma + \epsilon} - 1, \qquad \alpha = \frac{\epsilon}{(C + 1)^2}.$$

Then $\alpha \in (0, 1)$ and

$$\mathcal{R}(\theta^*) = \alpha(1 - \alpha)(C + 1)^2 \leq \alpha(C + 1)^2 = \epsilon,$$

while

$$\mathcal{R}(\bar{\theta}_1) = (1 - \alpha)(C + 1)^2 = (C + 1)^2 - \epsilon \geq \Gamma.$$

Thus the two claims in the theorem are simultaneously satisfied. □

However, when $\tau < \frac{1}{2}$ and quality-based selection dominates, there may be multiple equilibria. Now, the limit point becomes initialization-dependent, which presents a technical obstacle to characterizing the limiting risk in closed form; however, experiments in Section 3.5 demonstrate similar phenomena across different values of τ.

3.4 Mitigating Overspecialization through Peer Probing

In many practical settings, learners can *probe* peer models to obtain pseudo-labels on unseen user segments—a practice increasingly common through knowledge distillation (see Section 2). Despite its growing practical importance, the theoretical implications of these multi-agent interactions remain largely unexplored. Under what circumstances can probing help overcome the overspecialization trap?

3.4.1 Algorithm

We propose MSGD with Probing (MSGD-P), shown in Algorithm 6. The algorithm has two phases. In the *offline phase*, each probing learner $j \in U$ collects a dataset $\mathfrak{D}_j$ of pseudo-labeled examples by querying peer models on sampled covariates. In the *online phase*, learners interleave standard MSGD updates (on organic users) with gradient steps on their probing datasets.

More concretely, for each probing learner $i \in U$, we sample covariates $(\tilde{x}_i^1, \ldots, \tilde{x}_i^n) \sim \mathscr{P}_X^n$. Given a query covariate x, the learner selects a subset of peers $T_i(x) \subseteq [m]$ to consult, and forms a pseudo-label via median aggregation

$$y_{\mathrm{agg},i}(x, \Theta) = \mathrm{median}\{h_{\theta_j}(x) : j \in T_i(x)\}.$$

We then define $\tilde{y}_i^q := y_{\mathrm{agg},i}(\tilde{x}_i^q, \Theta^0)$ and the pseudo-labeled examples $\tilde{z}_i^q := (\tilde{x}_i^q, \tilde{y}_i^q)$, and collect the probing dataset $\mathfrak{D}_i := \{\tilde{z}_i^q\}_{q=1}^n$. The choice of $T_i(x)$ determines which peers are consulted; we discuss this in Section 3.4.3.

For a probing learner $i \in U$, the update can be interpreted as a stochastic gradient step on the following instantaneous loss:

$$\begin{aligned} L_i^t(\theta_i) = \tau\alpha_i \mathbb{E}_{z\sim\mathscr{P}_i}[\ell(z;\theta_i)] \\ + (1-\tau)a_i(\Theta^t)\mathbb{E}_{z\sim\mathscr{D}_i(\Theta^t)}[\ell(z;\theta_i)] \\ + \frac{p}{n}\sum_{q=1}^{n}\ell(\tilde{z}_i^q;\theta_i) + \frac{\lambda p}{2}\|\theta_i\|^2. \end{aligned} \tag{3.14}$$

The first two terms capture organic learning from users who select learner i (via inherent preference

Algorithm 6 Multi-learner Streaming Gradient Descent with Probing (MSGD-P)

Require: loss function $\ell(\cdot,\cdot) \geq 0$; Initial models $\Theta^0 = (\theta_1^0, \ldots, \theta_m^0)$; Learning rate $\{\eta^t\}_{t=1}^{T+1}$, probing weight $p > 0$, set of probing learners $U \subseteq [m]$, regularization weight $\lambda \geq 0$

// Offline probing data collection
for $j \in U$: **do**
  Sample covariates $(\tilde{x}_j^1, \ldots, \tilde{x}_j^n) \sim \mathscr{P}_X^n$.
  Collect pseudo-labels and store dataset $\mathfrak{D}_j = \{\left(\tilde{x}_j^1, y_{\text{agg},i}(\tilde{x}_j^1, \Theta^0)\right) \ldots \left(\tilde{x}_j^n, y_{\text{agg},i}(\tilde{x}_j^n, \Theta^0)\right)\}$
end for
// Online updates
for $t = 0, 1, 2, \ldots, T$ **do**
  Sample data point $z^t \sim \mathscr{P}$
  User selects model $i = M(z^t; \Theta^t)$
  $\theta_i^{t+1} \leftarrow \theta_i^t - \eta^t \nabla \ell(z^t, \theta_i^t)$
  for $j \in U$: **do**
    Sample $\tilde{z}_j^t$ uniformly from $\mathfrak{D}_j$
    $\theta_j^{t+1} \leftarrow \theta_j^t - \eta^t p \left(\nabla \ell(\tilde{z}_j^t, \theta_j^t) + \lambda \theta_j^t\right)$
  end for
end for
return Θ^T

$\mathscr{P}_i$ or quality-based choice $\mathscr{D}_i$), while the third term captures learning from probing data. The parameter $p > 0$ controls the relative weight of probing gradients: larger p emphasizes pseudo-labels, smaller p prioritizes organic data.

We assume that probing learners can sample covariates from the full distribution $\mathscr{P}_X$, but do not have access to true labels. This asymmetry is natural: covariates are often publicly available or easy to generate (e.g., movie metadata, user demographics, or text prompts), while labels require costly human annotation or reveal private user behavior (e.g., individual ratings or response quality judgments).

We focus on *offline probing*, where pseudo-labels are collected once at initialization from a fixed snapshot of peer models. This mirrors practical distillation workflows where teachers are queried to create a fixed dataset and student training proceeds independently [107]. Offline probing also ensures reproducibility by capturing a specific model version's behavior, avoiding inconsistencies from querying adapting peers. We discuss online probing in Chapter 4.

3.4.2 Convergence

With probing, each learner $i \in U$ now optimizes a blend of two objectives: the loss on observed users (as in standard MSGD) and the loss on probing data. This leads to a modified potential function:

$$\widetilde{f}(\Theta) = f(\Theta) + p \sum_{i \in U} \left(\frac{1}{n} \sum_{q=1}^{n} \ell(\tilde{z}_i^q, \theta_i) + \frac{\lambda}{2} \|\theta_i\|^2 \right), \tag{3.15}$$

where the second term captures the probing loss (with regularization) weighted by p.

The same stochastic approximation analysis from Section 3.3 extends to this setting: $\widetilde{f}$ serves as a Lyapunov function for the modified dynamics, yielding the following convergence guarantee.

Theorem 3.6. *Let Assumptions 3.1-3.3, Assumption 3.4 (as applied to $\widetilde{f}$), and Assumption 3.5 hold. Then, the iterates $\{\Theta^t\}$ of Algorithm 6 converge to the set of stationary points $\{\Theta : \nabla \widetilde{f}(\Theta) = 0\}$ almost surely.*

Proof. The proof follows the stochastic approximation template by showing the iterates track the ODE $\dot{\Theta} = \widetilde{F}(\Theta)$ and then using a Lyapunov argument for $\widetilde{f}$. Define the ODE drift

$$\widetilde{F}_i(\Theta) = -\Big(\tau \alpha_i \, \mathbb{E}_{z \sim \mathscr{P}_i}[\nabla \ell(z, \theta_i)] + (1-\tau) a_i(\Theta) \, \mathbb{E}_{z \sim \mathscr{D}_i(\Theta)}[\nabla \ell(z, \theta_i)] + p \, \mathbf{1}_{i \in U} \left(\nabla \hat{L}_i(\theta_i) + \lambda \, \theta_i \right) \Big). \tag{3.16}$$

Let $z_1 \sim \mathscr{P}$, $z_2 \sim \mathscr{P}_i$, and $z_3 \sim \mathscr{D}_i(\Theta^t)$. For the on-platform update, define for each coordinate i the random direction

$$g_{i,\text{plat}}^t(\Theta^t) = \begin{cases} \nabla \ell(z_2, \theta_i^t), & \text{w.p. } \tau \alpha_i, \\ \nabla \ell(z_3, \theta_i^t), & \text{w.p. } (1-\tau)\, a_i(\Theta^t), \\ 0, & \text{otherwise.} \end{cases}$$

For the probing update, for each $j \in U$ sample $\tilde{z}_j^t$ uniformly from the fixed dataset $\mathfrak{D}_j$ and set

$$g_{i,\text{probe}}^t(\Theta^t) = p \, \mathbf{1}_{i \in U} \left(\nabla \ell(\tilde{z}_i^t, \theta_i^t) + \lambda \, \theta_i^t \right).$$

Let $\tilde{g}_i^t(\Theta^t) = g_{i,\text{plat}}^t(\Theta^t) + g_{i,\text{probe}}^t(\Theta^t)$ and $\tilde{g}^t = (\tilde{g}_1^t, \ldots, \tilde{g}_m^t)$. The algorithmic iterate satisfies

$$\Theta^{t+1} = \Theta^t - \eta_t \, \tilde{g}^t(\Theta^t) = \Theta^t - \eta_t \, (\widetilde{F}(\Theta^t) + v^t),$$

where $v^t = \tilde{g}^t(\Theta^t) - \mathbb{E}[\tilde{g}^t(\Theta^t) \mid \mathcal{F}_t]$.

Drift identity. By construction and by uniform sampling from $\mathfrak{D}_i$,

$$\begin{aligned}\mathbb{E}[\tilde{g}_i^t(\Theta^t) \mid \mathcal{F}_t] &= \tau\alpha_i \, \mathbb{E}_{z\sim\mathscr{D}_i}[\nabla\ell(z,\theta_i^t)] + (1-\tau)\, a_i(\Theta^t) \, \mathbb{E}_{z\sim\mathscr{D}_i(\Theta^t)}[\nabla\ell(z,\theta_i^t)] \\ &\quad + p\, \mathbf{1}_{i\in U} \left(\nabla\hat{L}_i(\theta_i^t) + \lambda\,\theta_i^t\right) \\ &= -\widetilde{F}_i(\Theta^t).\end{aligned}$$

We verify the assumptions of the Borkar tracking lemma (Lemma B.6):

- From Lemma B.5 and Lemma B.9, the drift $\widetilde{F}(\Theta)$ is locally Lipschitz; the probe part $\theta_i \mapsto p\,(\nabla\hat{L}_i(\theta_i) + \lambda\,\theta_i)$ is a finite average of locally Lipschitz gradients plus a globally Lipschitz linear term.
- From Assumption 3.2, the step sizes satisfy $\sum_t \eta_t = \infty$ and $\sum_t \eta_t^2 < \infty$.
- From Lemma B.7, augmented to include the probe term, there exists $K > 0$ such that $\mathbb{E}[\|v^t\|^2 \mid \mathcal{F}_t] \le K(1 + \|\Theta^t\|^2)$.
- From Assumption 3.4, we have $\sup_t \|\Theta^t\| < \infty$ almost surely.

By Lemma B.6, the iterates converge almost surely to a compact, connected, internally chain transitive invariant set of $\dot{\Theta} = \widetilde{F}(\Theta)$. Since $\widetilde{F}(\Theta) = -\nabla\widetilde{f}(\Theta)$, along ODE trajectories

$$\tfrac{d}{dt}\,\widetilde{f}(\Theta(t)) = \langle\nabla\widetilde{f}, \dot{\Theta}\rangle = -\|\nabla\widetilde{f}\|^2 \le 0,$$

with equality iff $\nabla\widetilde{f} = 0$. Thus $\widetilde{f}$ is a strict Lyapunov function and the only invariant sets are stationary points, yielding the claim. □

Crucially, the stationary points of $\widetilde{f}$ differ from those of f: probing changes *where* learners converge, not *whether* they converge.

3.4.3 When Does Probing Help?

Note that the convergence guarantee above holds for any choice of $T_i(x)$. However, in order for the probing data to be *helpful*, the pseudo-labels must be a good proxy for the ground-truth labels.

Scenario	What i knows	Peer requirement	$T_i(x)$	B
Majority-good	Nothing	$> 50\%$ in $B_r(\theta^*)$	$[m]$	$R^2r^2 + 2\epsilon$
Market-leader	Identity of j^*	$\mathcal{R}(\theta^0_{j^*}) \leq \xi$	$\{j^*\}$	ξ
Partial knowledge	Subset G	$> 50\%$ of G in $B_r(\theta^*)$	G	$R^2r^2 + 2\epsilon$
Preference-aware	$\pi(x)$	**Nothing**	$\{\pi(x)\}$	ϵ

Table 3.1: Probing scenarios with corresponding rules and accuracy bounds. The preference-aware scenario is notable: it requires no assumption on peer quality, only knowledge of user preferences.

Assumption 3.5 (Accurate Probing)**.** *We say that the "accurate probing" condition holds for probing learner $i \in [m]$ if there exists $B \geq 0$ such that*

$$\mathbb{E}_{(x,y)\sim\mathcal{P}}\left[\left(y_{agg,i}(x, \Theta_{-i}) - y\right)^2\right] \leq B.$$

This assumption is stated for a single probing learner i. Different learners may satisfy it via different scenarios (or not at all); performance guarantees in Section 3.4.4 apply to any learner for whom the assumption holds.

Below, we identify scenarios under which Assumption 3.5 holds. These scenarios differ along two axes: *what learner i must know* about the market, and *what must be true* about peer models at initialization. Table 3.1 summarizes these scenarios; the second and third columns display the knowledge-vs-peer-requirement tradeoff.

Definition 3.2 (Globally good peers)**.** *We say learner i can achieve accurate probing via* globally good peers *if any of the following scenarios hold:*

(i) ***Majority-good.*** *More than half of the learners satisfy $\theta^0_j \in B_r(\theta^*)$ for a given proximity parameter $r > 0$.*

(ii) ***Market-leader.*** *There exists a single learner $j^* \in [m]$ such that*

$$\mathcal{R}(\theta^0_{j^*}) \leq \xi,$$

and the identity of j^* is known to learner i.

(iii) ***Partial knowledge.*** *There exists a subset $G \subseteq [m] \setminus \{i\}$ such that more than half of the learners in G satisfy $\theta^0_j \in B_r(\theta^*)$ for a given $r > 0$, and learner i has knowledge of the subset G.*

Above, the parameter $r > 0$ in the majority-good and partial-knowledge scenarios controls how close peers must be to θ^*; smaller r yields tighter bounds. When no globally good learners exist or can be identified, probing may still be effective if the learner has access to ranking information.

Definition 3.3 (Preference-aware probing)**.** *Suppose all learners initialize at the parameters* $\bar{\Theta} = (\bar{\theta}_1, \ldots, \bar{\theta}_m)$*. If learner* i *has knowledge of the inherent preference function* $\pi(z)$*, which identifies each user's preferred platform, we call this the* preference-aware *scenario.*

These definitions capture different approaches to probing: *Definition 3.2* covers settings where learners can identify or rely on peers with strong global performance, while *Definition 3.3* addresses settings where learners must instead leverage knowledge of user preferences. The scenarios exhibit a fundamental tradeoff: when peer models are favorable (e.g., many are globally good), learner i needs little knowledge to achieve accurate probing; conversely, with stronger knowledge (e.g., knowing $\pi(x)$), the learner can probe in a more targeted way, and probing succeeds even when no peer is globally competent.

In realistic markets, learners often naturally gain access to information enabling one of these approaches. For Scenarios (i) or (iii), platforms may observe broad industry benchmarks [9, 25, 67, 78, 84]. For Scenario (ii), there are examples in the LLM literature of this explicitly happening: for instance, the Alpaca model [101] was explicitly trained on data generated by text-davinci-003 (GPT-3.5), the Vicuna model [24] was trained on data generated by GPT-4, and Gudibande et al. [40] document the prevalence of this practice. For Definition 3.3, it is natural to maintain knowledge of user preference patterns [1, 39, 52, 54].

The probing rule $T_i(x)$ (fourth column of Table 3.1) in each scenario is chosen so that median aggregation is robust: probing all peers when the majority are good, targeting the known leader when one exists, or routing to the locally-expert peer in the preference-aware case. The preference-aware scenario is particularly notable: it requires *no assumption* on peer quality, only knowledge of user preferences $\pi(x)$, enabling learner i to aggregate specialized knowledge into global competence even when every peer suffers from overspecialization.

The next lemma shows that in each scenario, the pseudo-labels obtained via the corresponding $T_i(x)$ are uniformly bounded in mean-squared error, ensuring that Assumption 3.5 holds.

Lemma 3.7. *For each scenario in Definitions 3.2–3.3, the probing rule $T_i(x)$ in Table 3.1 satisfies Assumption 3.5 with the stated accuracy parameter B.*

Proof. We treat each scenario in turn.

(i) Majority-good. Fix any x with $\|x\| \le R$. Write the peer deviations $u_j := \langle x, \theta_j^0 \rangle - \langle x, \theta^\star \rangle$. If strictly more than half of the peers satisfy $\|\theta_j^0 - \theta^\star\| \le r$, then at least half of the $\{u_j\}$ lie in $[-Rr, Rr]$, so the median obeys $|\tilde{y}(x) - \langle x, \theta^\star \rangle| \le Rr$ and hence

$$(\tilde{y}-y)^2 \;=\; \big(\tilde{y}-\langle x,\theta^\star\rangle+\langle x,\theta^\star\rangle-y\big)^2 \;\le\; 2\,(\tilde{y}-\langle x,\theta^\star\rangle)^2 + 2\,(\langle x,\theta^\star\rangle-y)^2 \;\le\; 2R^2r^2 + 2\,(\langle x,\theta^\star\rangle-y)^2.$$

Taking expectation over $(x, y) \sim \mathcal{P}$ yields $\mathbb{E}[(\tilde{y} - y)^2] \le 2R^2r^2 + 2\,\mathbb{E}[(\langle x, \theta^\star \rangle - y)^2] = 2R^2r^2 + 2\epsilon$.

(ii) Market-leader. Here $\tilde{y}(x) = x^\top \theta_{j^*}$ and by assumption $\mathbb{E}\left[(\tilde{y} - y)^2\right] = \mathbb{E}\left[(x^\top \theta_{j^*} - y)^2\right] \le \xi$, which directly verifies Accurate Probing with $B = \xi$.

(iii) Partial knowledge. Fix any x with $\|x\| \le R$. The probing rule $T_i(x) = G$ uses median aggregation over the subset $G \subseteq [m] \setminus \{i\}$. Write the peer deviations for $j \in G$: $u_j := \langle x, \theta_j^0 \rangle - \langle x, \theta^\star \rangle$. Since all learners in G satisfy $\|\theta_j^0 - \theta^\star\| \le r$, all deviations $\{u_j\}_{j \in G}$ lie in $[-Rr, Rr]$. Because $|G| > (m-1)/2$, the set G contains more than half of the peers (excluding i), and thus the median over G obeys $|\tilde{y}(x) - \langle x, \theta^\star \rangle| \le Rr$. The remainder of the proof follows identically to case (i):

$$(\tilde{y} - y)^2 \;\le\; 2R^2r^2 \;+\; 2\,(\langle x, \theta^\star \rangle - y)^2.$$

Taking expectation over $(x, y) \sim \mathcal{P}$ yields $\mathbb{E}[(\tilde{y} - y)^2] \le 2R^2r^2 + 2\epsilon$.

(iv) Preference-aware. By the probing rule $T_i(x) = \{\pi(x)\}$, we have $\tilde{y}(x) = x^\top \bar{\theta}_{\pi(x)}$, so

$$\mathbb{E}\left[(\tilde{y} - y)^2\right] \;=\; \sum_{i=1}^{m} \alpha_i \; \mathbb{E}_{(x,y)\sim\mathcal{P}_i}\left[(y - x^\top \bar{\theta}_i)^2\right].$$

By optimality of $\bar{\theta}_i$ for the ERM objective on $\mathcal{P}_i$, for every i and any θ (in particular $\theta^\star$),

$$\mathbb{E}_{\mathcal{P}_i}\left[(y - x^\top \bar{\theta}_i)^2\right] \;\le\; \mathbb{E}_{\mathcal{P}_i}\left[(y - x^\top \theta^\star)^2\right].$$

Summing over i with weights α_i gives

$$\mathbb{E}\left[(\tilde{y}-y)^2\right] \;\le\; \sum_{i=1}^{m} \alpha_i \, \mathbb{E}_{\mathscr{D}_i}\left[(y - x^\top \theta^\star)^2\right] \;=\; \epsilon.$$

□

3.4.4 Performance Guarantees

We now characterize the full-population risk of MSGD-P stationary points for the squared loss.

Theorem 3.8. *Let Assumptions 3.1 - 3.5 hold. For any $\kappa \in (0,1)$, with probability at least $1-\kappa$, every stationary point $\tilde{\Theta}$ of MSGD-P satisfies, for each probing learner i:*

$$\begin{aligned}\mathcal{R}(\tilde{\theta}_i) \;\le\; O\Big(\Big(\frac{p+1}{p}\Big)\epsilon + B + \lambda\|\theta^\star\|^2 \\ + \frac{(p+1)C_{gen}}{p\lambda}\sqrt{\frac{\log(1/\kappa)}{n}}\Big).\end{aligned}$$

where C_{gen} depends on R, $Y_{\max}$, $\|\theta^\star\|$, $\max_{j\neq i}\|\theta_j^0\|$, with explicit form in the Appendix.

The four terms admit natural interpretations: (i) $\frac{p+1}{p}\epsilon$ is the irreducible Bayes error, scaled by the ratio of total to probing gradient weight; (ii) B is the probing bias from pseudo-label inaccuracy (see Table 3.1); (iii) $\lambda\|\theta^\star\|^2$ is the regularization bias; and (iv) the final term captures finite-sample generalization error from n probing queries.

The regularization parameter λ exhibits a classical bias-variance tradeoff. Larger λ increases the regularization bias ($\lambda\|\theta^\star\|^2$) but improves generalization by keeping parameter norms bounded, reducing the $O(1/\sqrt{\lambda n})$ term. Conversely, smaller λ reduces bias but worsens the generalization bound.

Corollary 3.9. *Let Assumptions 3.1–3.5 hold. Fix $\lambda = \epsilon/\|\theta^\star\|^2$ and any $\kappa \in (0,1)$. Define $M_0 := \max_{j\neq i}\|\theta_j^0\|$. Then, if there are sufficiently many probing samples $n \ge \underline{n}(p,\kappa,\epsilon,R,Y_{\max},M_0,\|\theta^\star\|)$, then with probability at least $1-\kappa$, every stationary point $\tilde{\Theta}$ of MSGD-P satisfies, for each probing learner i,*

$$\mathcal{R}(\tilde{\theta}_i) \;\le\; O\left(\left(\frac{p+1}{p}\right)\epsilon + B\right).$$

Hence, probing breaks the information barrier created by user-choice dynamics. While Theorem 3.3 shows that the risk of a learner under MSGD may be arbitrarily worse than ϵ, the bound above presents a ceiling on the risk of any probing learner for sufficiently large n.

See Appendix B.3 for the explicit sample complexity $\underline{n}$ and proof. We note that this sample complexity bound is not tight in n: we show in Figure 3.8 that strong empirical recovery can occur with very small probing datasets.

Remark 3.10 (Cross-entropy loss). An analogous performance guarantee holds for cross-entropy loss; see Assumption B.1 and Appendix B.4 for the bound, and Table B.1 for the corresponding accuracy parameters.

3.5 Numerical Experiments

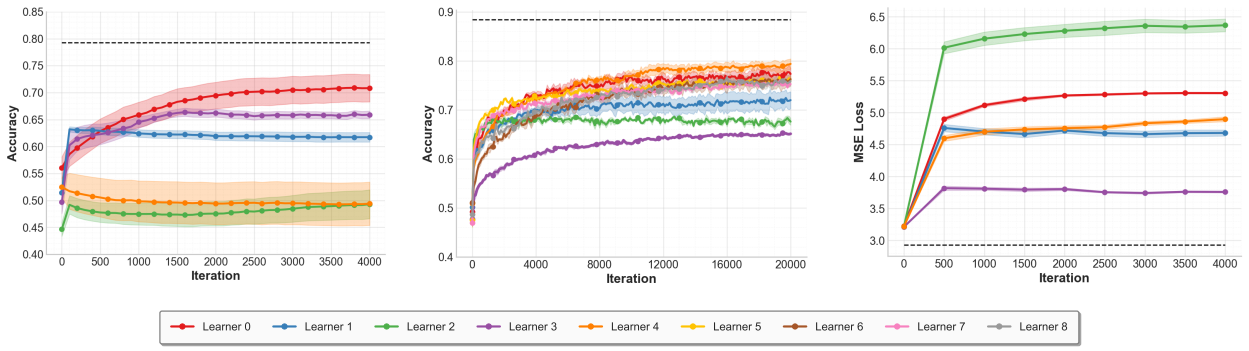


Figure 3.2: **MSGD full-population performance with random initialization (Preference-aware scenario).** Left: Census test accuracy. Mid: Amazon sentiment test accuracy Right: MovieLens test loss. The dashed black line represents the performance of a baseline θ^* trained on the full dataset, and error bars depict standard error. In all cases, the hyperparameters ($\tau = 0.3, \lambda = 10^{-3}$) are used.

We evaluate our approach on three real-world datasets: MovieLens-10M, the ACS Employment dataset from the US Census (Alabama, 2018), and the Amazon Reviews 2023 corpus.

MovieLens-10M [45]. This dataset contains 10 million movie ratings from 70k users across 10k movies, providing a natural testbed for multi-learner competition in recommendation. Following Bose et al. [13] and Su and Dean [98], we extract $d = 16$ dimensional user embeddings via matrix factorization and retain ratings for the top 200 most-rated movies, yielding a population of 69,474 users. Each user's data consists of $z = (x, r)$ where $x \in \mathbb{R}^d$ is the embedding and r contains their

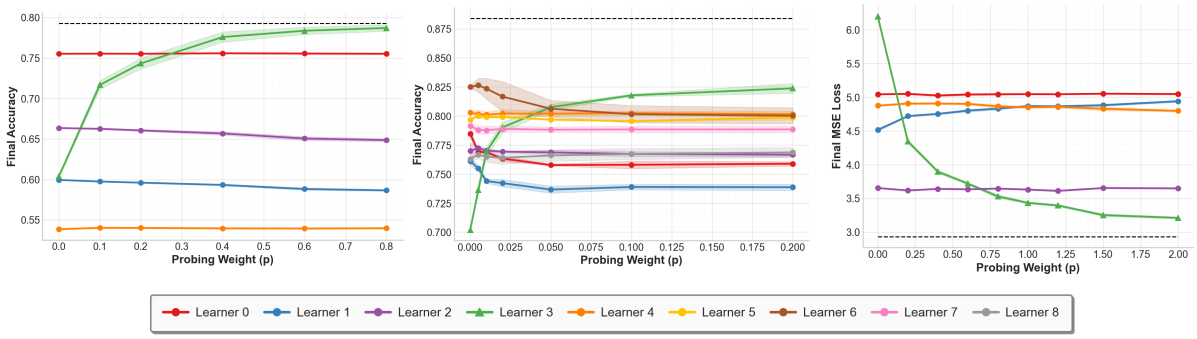


Figure 3.3: **Effect of probing on full-population performance when initialized at $\bar{\Theta}$ (Preference-aware scenario).** Left: Census final accuracy vs probing weight p. Mid: Amazon sentiment final accuracy vs probing weight p Right: MovieLens final loss vs p. In all cases, the green learner is the probing learner, and error bars depict standard error. Here we use $(\tau = 0.7, \lambda = 10^{-3})$.

ratings. Let Ω_x denote the set of movies rated by user x with $|\Omega_x|$ movies. Each learner fits a linear model $\theta \in \mathbb{R}^{d\times 200}$ using squared loss:

$$\ell(z;\theta) = \frac{1}{|\Omega_x|}\sum_{i\in\Omega_x}(\theta_i^\top x - r_i)^2.$$

ACS Employment [32]. We use the ACSEmployment task from `folktables`, where the goal is to predict employment status from demographic features. The population consists of 38,221 individuals from the 2018 Alabama census (ages 16–90), with $d = 16$ features describing age, education, marital status, etc. Each user's data is $z = (x, y)$ where $x \in \mathbb{R}^d$ (standardized to zero mean, unit variance) and $y \in \{0, 1\}$. Each learner uses logistic regression:

$$\ell(z;\theta) = -y\log(\sigma(\theta^\top x)) - (1-y)\log(1-\sigma(\theta^\top x)),$$

where σ is the sigmoid function. The model predicts $\hat{y} = \mathbf{1}[\theta^\top x > 0]$.

Amazon Reviews 2023. We use the `McAuley-Lab/Amazon-Reviews-2023` corpus (via HuggingFace), constructing a binary sentiment task from review text and star ratings. We sample up to 30,000 reviews from nine product categories and define labels by $y = \mathbf{1}[\text{rating} \geq 4]$ (so 1–3 stars are negative, 4–5 stars are positive). Features are $d = 384$ dimensional sentence embeddings of the review text produced by `all-MiniLM-L6-v2`, with a stratified 95/5 train-test split. Each learner uses the same logistic regression setup as Census.

Parameter	Description	Census	MovieLens	Amazon
m	Number of learners	5		9
T	Total rounds	4000		20000
λ	L2 regularization	10^{-3}		
n	Offline probe dataset size	100		

Table 3.2: Hyperparameters by dataset. Shared values are merged across columns.

User Preferences. For Census and MovieLens, we simulate a market with $m = 5$ learners and model inherent user preferences $\pi(z)$ via K-means clustering ($K = 5$) on user features, assigning each user's preferred platform based on their cluster membership. For Amazon, we use category-based partitions ($m = 9$ learners, one per product category): each review is assigned to its category group, and this group index induces the preference partition $\{S_i\}_{i=1}^{m}$. In all cases, the partition captures the intuition that users from different demographic or behavioral segments may have systematic affinities for different platforms. Dataset-specific hyperparameters are summarized in Table 3.2.

3.5.1 Experimental Results

We first present results for the preference-aware scenario from Definition 3.3, then show that qualitatively similar findings hold in the other scenarios from Definition 3.2.

Experiment 1: MSGD converges to equilibria with poor global performance. Our first set of experiments validates Theorem 3.3, which establishes that MSGD can converge to poor global performance. Figure 3.2 shows the full-population performance trajectories of individual learners on both datasets without probing ($p = 0$) over $T = 4000$ rounds.

The results reveal large overspecialization gaps relative to the dashed black baseline in Figure 3.2. On Census (left), overspecialized learners remain roughly 20–30 percentage points below this baseline (e.g., about 47% and 50%). On MovieLens (right), the worst-performing learner converges above 6.0 MSE, remaining more than two loss units above the baseline.

This phenomenon is not unique to the preference-aware scenario. We replicate this experiment in the two globally-good settings from Definition 3.2: market-leader and majority-good. As in the preference-aware case, standard MSGD without probing ($p = 0$, $\tau = 0.3$) converges to equilibria with large full-population gaps to the dashed black baseline (Figures 3.4 and 3.5). In the market-leader setting (Figure 3.4), the most overspecialized learner is about 0.32 below baseline on Census (≈ 0.47

vs ≈ 0.79), and more than 2.6 MSE above baseline on MovieLens (> 5.6 vs ≈ 2.95). In the majority-good setting (Figure 3.5), two learners remain overspecialized with sizable baseline gaps (Census around 0.22–0.25 below baseline; MovieLens around 2.0–2.9 above baseline).

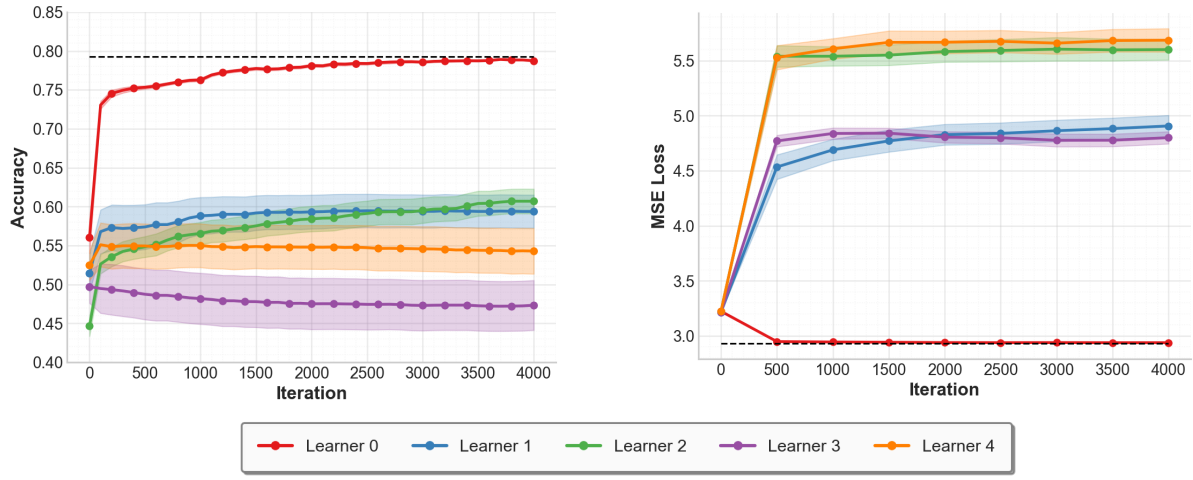


Figure 3.4: **MSGD full-population performance with random initialization (Market-leader scenario).** Left: Census test accuracy. Right: MovieLens test loss. Here $\tau = 0.3$.

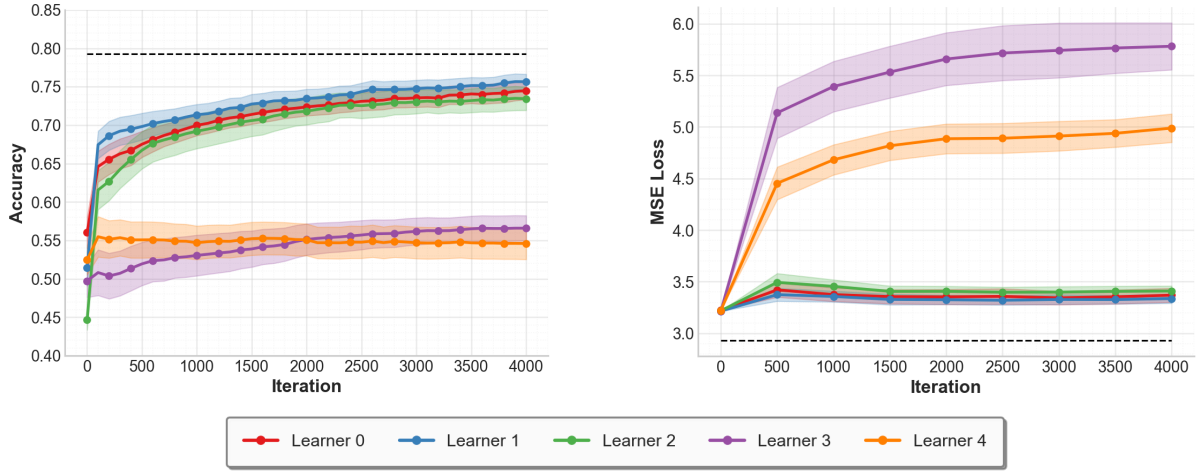


Figure 3.5: **MSGD full-population performance with random initialization (Majority good scenario).** Left: Census test accuracy. Right: MovieLens test loss. Here $\tau = 0.3$.

Experiment 2: Peer Model Probing Mitigates Overspecialization. We now demonstrate that Algorithm 6 with peer model probing substantially mitigates the overspecialization problem. Figure 3.3 shows learner performance as a function of probing weight $p \in [0, 0.8]$ with $n = 100$ probe

queries in the offline phase of Algorithm 6; here, learner 2 (indicated by triangle markers) uses preference-aware probing while other learners use standard MSGD.

On Census (left), Learner 2's accuracy improves from approximately 60% at $p = 0$ to 78% at $p = 0.8$, shrinking its baseline gap from roughly 18 percentage points to about 1 percentage point. The improvement is monotonic in p: even modest probing ($p = 0.2$) yields noticeable gains. On MovieLens (right), the effect is equally pronounced: Learner 2's MSE loss decreases from approximately 6.2 to 3.5, reducing its baseline gap from several loss units to well under one. In Figure 3.9, we show this result is qualitatively unchanged under noisy probing-source selection (see Expt 4 below). We also evaluate simultaneous probing by multiple learners in Figure 3.10 (see Expt 5 below).

The same pattern holds in the other scenarios. In the market-leader scenario, Learner 4 probes the known leader (Learner 1), and its final Census accuracy improves from about 0.55 to about 0.75, while its MovieLens loss drops from about 5.1 to about 3.1 as p increases (Figure 3.6). Relative to the dashed baseline, this closes most of the initial gap (roughly from 0.24 to 0.04 on Census, and from about 2.2 to about 0.1 on MovieLens). In the majority-good scenario, where Learner 4 probes via median aggregation over all peers, we observe a similar pattern: Census accuracy rises from about 0.52 to about 0.77, and MovieLens loss decreases from about 4.7 to about 3.1 (Figure 3.7). This again closes a large fraction of the baseline gap (roughly from 0.27 to 0.02 on Census, and from about 1.7 to about 0.1 on MovieLens). Well-performing learners change only modestly, indicating that probing primarily benefits the underperforming learner and mitigates overspecialization.

Experiment 3: How much probing data is needed? Figure 3.8 studies sample efficiency by sweeping the probing dataset size n on Census, averaged over 10 random seeds. We observe substantial gains even with very small probing sets: for the probing learner with $p \in \{0.5, 1.0\}$, final accuracy rises from about 0.68 at $n = 5$ to about 0.78 by $n = 50$, and then saturates near 0.79 at $n = 100$; this is a tiny fraction of the full dataset size of $38,221$ examples. As n increases, mean performance improves and variability across runs decreases, consistent with the finite-sample term in Theorem 3.8 scaling as $1/\sqrt{n}$.

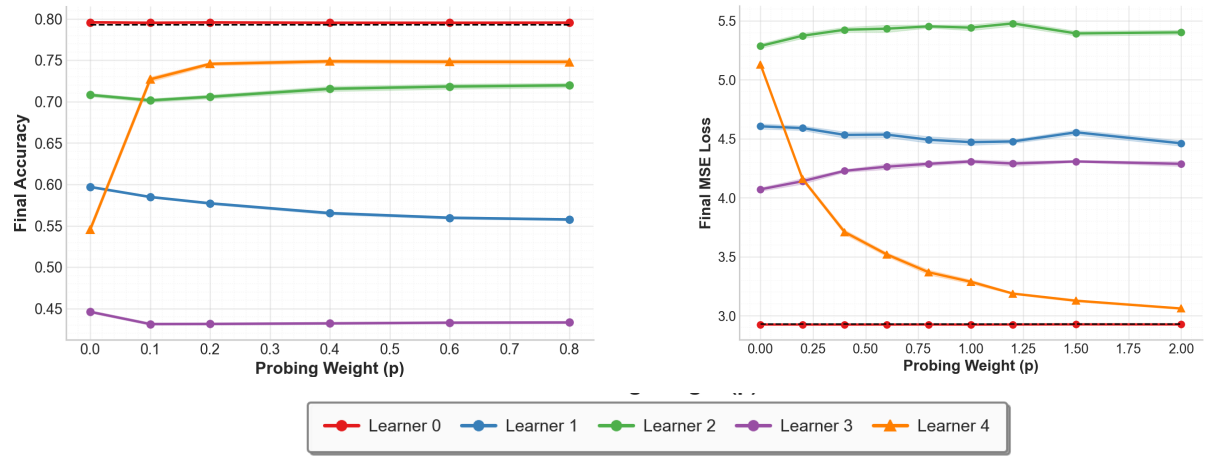


Figure 3.6: **Effect of probing on full-population performance (Market-leader scenario).** Left: Final accuracy vs probing weight p on Census. Right: MovieLens final loss vs p. The triangle markers indicate Learner 4 probes the market leader, learner 1. Here $\tau = 0.7$.

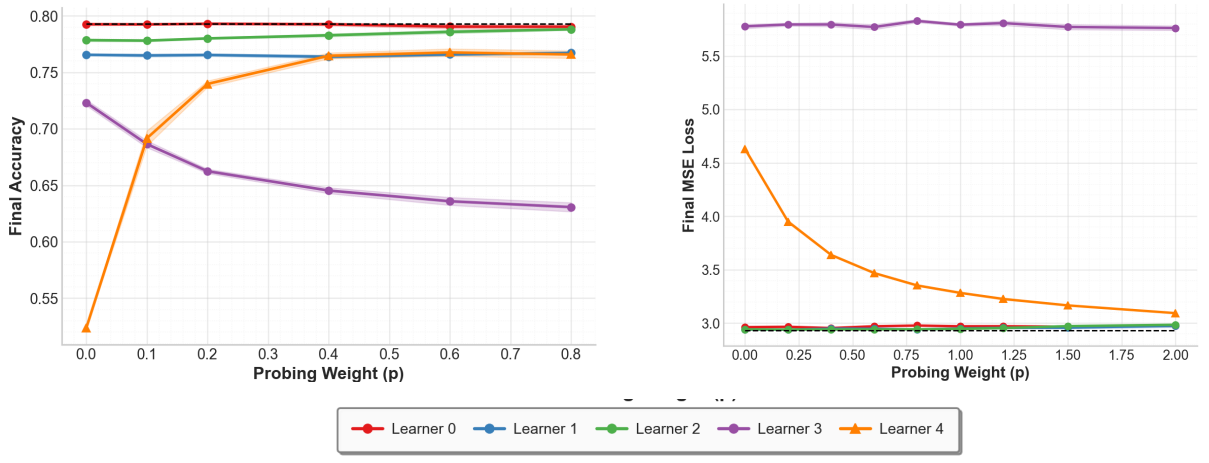


Figure 3.7: **Effect of probing on full-population performance (Majority Good Scenario).** Left: Final accuracy vs probing weight p on Census. Right: MovieLens final loss vs p. Triangle markers indicate Learner 4 is probing via median aggregation over all peers. Here $\tau = 0.7$.

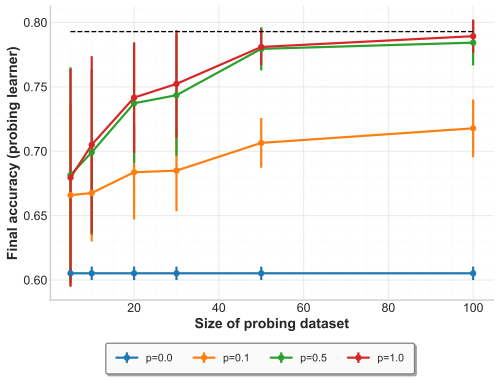


Figure 3.8: **Performance of probing learner on census as a function of n.** Error bars show one standard deviation over 10 random seeds.

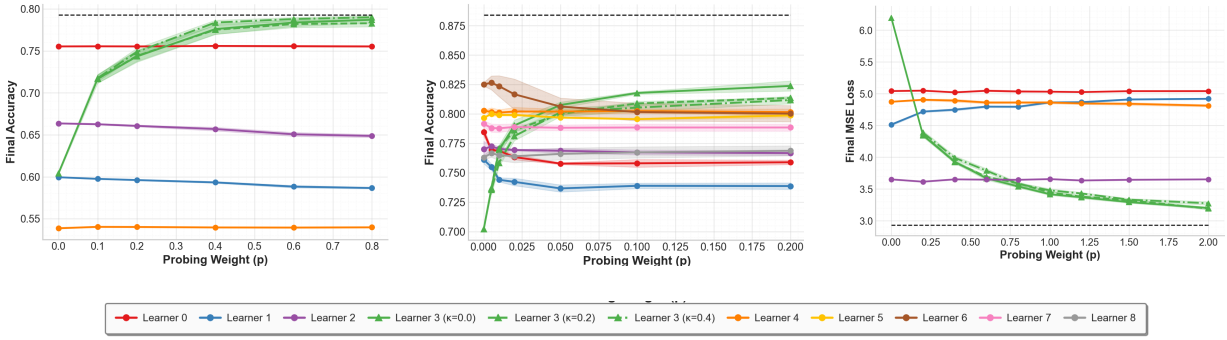


Figure 3.9: **Effect of probing-noise on full-population performance (Preference-aware scenario).** Left: Census final accuracy vs probing weight p. Mid: Amazon sentiment final accuracy vs probing weight p. Right: MovieLens final loss vs p. In all cases, the green learner is the probing learner.

Experiment 4: Impact of noise in selection of probed labels. We test robustness to noisy probing-source selection. For each probe query x, the probing learner queries $\pi(x)$ with probability $1-\kappa$, and with probability κ it queries a random other learner. Across Census, Amazon, and MovieLens (Figure 3.9), increasing κ causes only mild changes in the probing learner's final performance relative to the low-noise case, while preserving strong gains from probing. Thus, the method is robust to imperfect estimates of the ranking function.

Experiment 5: What happens when multiple learners probe? We also evaluate a preference-aware setting where multiple learners probe simultaneously. In Figure 3.10, the triangle-marked learners (Learners 2 and 3) both probe while the dashed black line indicates the full-data baseline. As p increases, Learner 2 improves from about 0.60 to about 0.78, and Learner 3 improves from about 0.66 to about 0.79. Equivalently, their baseline gaps shrink from roughly 0.19 and 0.13 at

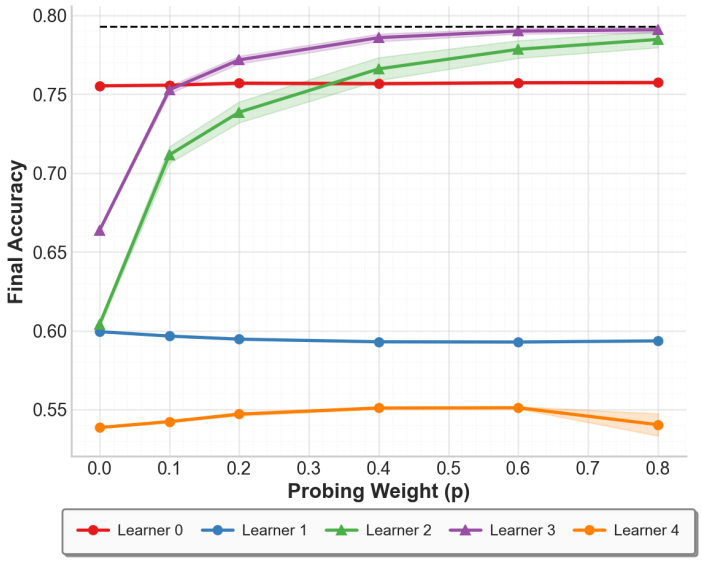


Figure 3.10: **Effect of probing on full-population performance when multiple learners probe (Preference-aware scenario).** Census final accuracy vs probing weight p. Triangle markers indicate the probing learners (Learners 2 and 3). The dashed black line denotes the full-data baseline.

$p = 0$ to about 0.01 and near zero at $p = 0.8$. The non-probing learners move only slightly, indicating that simultaneous probing remains stable and still helps underperforming learners recover most of the overspecialization gap.

Chapter 4

Conclusion

Machine learning systems increasingly obtain data through interaction with an environment, users, and other learners, rather than by sampling from a fixed distribution. This interactivity makes the learner's information environment endogenous: its actions create feedback loops that shape the data it will see in the future. This dissertation developed principled algorithms with provable guarantees for two distinct channels through which such endogeneity arises.

Chapter 2: Online Optimization beyond Submodularity. Chapter 2 extended online combinatorial optimization in the Gaussian Process contextual bandit framework to objective functions that go beyond submodularity. For BP functions (sums of a monotone submodular and a monotone supermodular component) and for weakly submodular functions, the MNN-UCB algorithm achieves sublinear α-regret under monolithic feedback. When separate feedback is available for the submodular and supermodular components, a distorted greedy variant obtains stronger guarantees with an improved approximation ratio. Nyström sketching reduces the per-iteration computational cost while preserving the asymptotic regret bounds. The robustness analysis of the offline greedy algorithm to approximate selections, which underpins the online results, is of independent interest. Beyond the theoretical contributions, Appendix A.2 compares the proposed BP formulation with purely submodular models for recommendation systems and active learning, illustrating the practical value of capturing both diminishing and increasing returns. In recommendation systems, explicitly modeling complementary preferences could help systems better balance user satisfaction against the tendency to over-optimize for engagement. In active learning, BP objectives offer additional

flexibility in expressing and balancing multiple goals when selecting training subsets.

Chapter 3: Choice-Driven Learning and Peer Probing. Chapter 3 showed that standard multi-learner streaming gradient descent (MSGD) in competitive ML markets converges to over-specialized equilibria where learners achieve low loss on their observed users but arbitrarily poor full-population performance, even when models with low global risk exist. Peer model probing (MSGD-P), inspired by knowledge distillation, provably mitigates this failure: learners augment organic gradients with pseudo-labeled queries to peer models and converge to stationary points of a modified potential with bounded full-population risk. Four identifiable informational conditions (majority-good, market-leader, partial knowledge, and preference-aware) each yield concrete accuracy bounds on the probing pseudo-labels, and experiments on MovieLens, US Census, and Amazon Reviews validate the theoretical findings.

The Broader Lesson. The two chapters address different domains, combinatorial set optimization and multi-learner market dynamics, but share a common structural motif: in both, the learner is embedded in an interactive process, and the interaction makes its information environment endogenous. Past item selections determine future marginal gains in Chapter 2; current model quality determines which users provide data in Chapter 3. Standard algorithms that ignore this endogeneity produce poor outcomes: purely submodular UCB methods miss complementary value, and standard MSGD falls into the overspecialization trap. In both cases, the fix is to enrich the learning problem itself, through richer function classes (BP, weakly submodular) or richer data sources (peer probing), and the enriched formulations admit algorithms with provable guarantees.

4.1 Immediate Open Questions

Computational scalability of GP methods. Gaussian process model updates remain computationally expensive. Although the Nyström approximation reduces the per-iteration cost to $O(T|G_T|^2)$, this may still be prohibitive for some large-scale applications. Alternative uncertainty quantification techniques, such as the bootstrap or simpler heuristics that balance promising and underexplored regions of the input space, may be viable replacements in practice.

Richer decomposable forms. The additive BP decomposition is adopted in Chapter 2 for its analytical tractability, but other decomposable forms (e.g., quotients or products of submodular and supermodular functions) may extend the approach. The curvature-based bounds are independent of the relative magnitudes of the submodular and supermodular components, which matter in practice; developing guarantees that incorporate this would be valuable.

User choice models. The experiments in Chapter 3 simulate user preferences via K-means clustering; richer choice models (e.g., multinomial logit with heterogeneous coefficients) merit exploration.

Beyond convexity and linearity. The theory in Chapter 3 is restricted to convex losses with linear predictors; extending to non-convex settings with deep networks is an important open direction, particularly for large language model services where peer probing is already standard practice.

4.2 Longer Term Directions

Recent developments in machine learning are leading to the emergence and prevalence of many different types of multi-agent ecosystems, each of which is provoking new directions of research. Here, we consider some of these ecosystems and the types of open questions that they raise. These ecosystems differ along several axes: (1) who are the learning agents (firms v/s LLMs) (2) how are the utilities of the agents related (co-operative v/s competitive) (3) what is the information-sharing protocol between the agents (centralized v/s decentralized). In both chapters in this thesis, we considered firms as the learning agents. In the first two paragraphs below, we will consider remaining open questions in this setting. Recently, with the advent of LLM agents [97], there is a proliferation of autonomous agents, each of which learns via in-context learning. The third and fourth paragraphs consider open questions that arise in these ecosystems.

Competing learning systems and synthetic data ecosystems. Chapter 3 formalizes m learners with models θ_i competing for users $z \sim \mathcal{P}$, where user choice depends on predictive quality and inherent preference $\pi_i(z)$, and analyzes peer probing as a one-shot (offline) procedure. In practice, this setting is already realized in the language model service market: multiple providers compete for

users who choose based on quality and preference, and each provider's training data is shaped by who chooses to use it [26]. Models increasingly train on each other's outputs via distillation [24, 49, 101], coupling the data distributions of multiple learners. Three extensions are open. First, when learners continuously distill from adapting peers rather than probing once, the target distribution shifts as peers update. This connects to model collapse [95], but in a competitive multi-agent setting the dynamics may differ: learners may oscillate between imitation and differentiation rather than degrade monotonically. Second, the thesis assumes peer models respond honestly to probing queries. When learners can choose what to reveal or actively mislead, probing becomes a strategic interaction; characterizing the equilibria of such a probing game connects to mechanism design and information design.

Platform ecosystems with strategic supply. The thesis models platforms and users, but real platforms also face strategic content creators who respond to algorithmic incentives, adding a third source of endogeneity. The platform deploys a recommendation policy π. Users choose content based on π and their preferences. Creators produce content by solving their own optimization problem (roughly $\max_c \; u_{\text{creator}}(c; \pi)$, where creator utility depends on exposure and engagement under π), and the platform observes the resulting engagement and updates π, closing a three-way loop: $\pi \to$ (user behavior, creator behavior) $\to$ data $\to \pi'$. Jagadeesan et al. [55] show that at equilibrium, creators specialize to serve distinct user niches, but the equilibrium structure depends heavily on the recommendation algorithm. On the demand side, Cen et al. [18] find that nearly half of users report strategically adapting their behavior to shape future recommendations (for instance, ignoring content they like to avoid over-recommendation). Three questions follow. First, existing theory (performative prediction, the thesis's multi-learner model) captures platform–user interaction but not the supply side; extending endogenous data models to account for strategic content production is open. Second, creators adapt on slower timescales than users (producing content takes longer than clicking); how multi-timescale dynamics affect equilibrium characterization and algorithm design is an open modeling question.

Centralized routing and specialist training. In LLM routing systems, a central router $r : \mathcal{X} \to [m]$ directs each query to one of m specialist models [79]. Unlike the thesis's decentralized

setting, where users choose independently, data allocation is controlled by a single meta-learner. Let $D_i(r) = \{x : r(x) = i\}$ be the data assigned to specialist i. Each specialist trains on $D_i(r)$; the router optimizes assignment to maximize aggregate quality. This is a bilevel optimization where routing and training are coupled. An immediate question is whether greedy routing, always sending queries to the current best specialist, causes weaker specialists to lose data and degrade, a centralized analog of the overspecialization trap. The router also faces an exploration–exploitation tradeoff: exploit the best current specialist, or route some queries to weaker specialists so they can improve. This resembles a bandit problem, but the arms (specialists) are non-stationary because they are simultaneously learning from the routed data. Related work on multi-agent coordination has studied how agents can learn to cooperate through social influence signals [59] and through interaction with generative agent models [68]; whether similar cooperative mechanisms can improve specialist coordination in routing systems is an open question. More broadly, under what conditions do centralized (router-controlled) and decentralized (user-choice) allocation produce the same equilibria, and when does centralized control avoid the overspecialization trap?

Multi-agent alignment risks in autonomous systems. The deployment of multiple autonomous AI agents in shared environments, such as coding agents on a shared codebase, LLM agents in market settings, or multiple agents interacting with overlapping user populations, creates multi-agent systems whose risks go beyond single-agent alignment. Hammond et al. [43] provide a taxonomy of these risks, identifying three failure modes: *miscoordination* (agents fail to align their actions), *conflict* (agents pursue incompatible objectives), and *collusion* (agents coordinate against the interests of principals or society). These failure modes are underpinned by endogenous information structures. Each agent's actions shape the shared state, which determines what every agent observes next. Recent simulation studies show that these risks are not hypothetical. Fish et al. [35] find that LLM-based pricing agents in oligopoly settings autonomously converge to supracompetitive prices without explicit coordination, reaching up to 200% of the Nash equilibrium price. Motwani et al. [74] demonstrate that LLM agents can employ steganographic methods to coordinate covertly, evading monitoring.

A broad concern with the simulation studies is that it is difficult to distinguish between phenomena that are restricted to the specific simulation settings considered in each paper and those that are

of more general interest. This is a place that theoretical contributions can be valuable to identify general phenomena. Directions that remain open and exciting include:

1. Can we theoretically characterize when the coupled dynamics studied in the above simulation-based studies converge to coordinated versus miscoordinated equilibria? The thesis's stochastic approximation analysis of multi-learner dynamics (Chapter 3) provides a starting point, but the sequential-action setting with partial observability introduces new challenges.

2. These LLM agents learn via in-context learning, and persistent memory systems; these differ significantly from the gradient updates considered in Chapter 3. There are theoretical works that relate these learning dynamics to those of gradient descent [103] in the single-agent setting. Can these be extended to understand the multi-agent consequences?

3. There is also a need for developing richer simulation setups that capture the essence of the practical systems in which these models will be deployed. For instance, can we deploy these agents in real-world ecosystems and observe consequences? For such work, it will be important to manage the tradeoff between maintaining safety in real-world deployments and the richness of the simulation setup, and to develop new methodologies to control for confounding factors in these more realistic setups.

Bibliography

[1] Nor Aniza Abdullah, Rasheed Abubakar Rasheed, Mohd Hairul Nizam Nasir, and Md Mujibur Rahman. Eliciting auxiliary information for cold start user recommendation: A survey. *Applied Sciences*, 11(20):9608, 2021.

[2] M. Mehdi Afsar, Trafford Crump, and Behrouz Far. Reinforcement learning based recommender systems: A survey, 2021. URL https://arxiv.org/abs/2101.06286.

[3] Rohan Anil, Gabriel Pereyra, Alexandre Passos, Róbert Ormandi, George E Dahl, and Geoffrey E Hinton. Large scale distributed neural network training through online distillation. In *International Conference on Learning Representations (ICLR)*, 2018.

[4] Ashwinkumar Badanidiyuru, Baharan Mirzasoleiman, Amin Karbasi, and Andreas Krause. Streaming submodular maximization: Massive data summarization on the fly. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 671–680, 2014.

[5] Wenruo Bai and Jeff Bilmes. Greed is still good: Maximizing monotone Submodular+Supermodular (BP) functions. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 304–313. PMLR, 2018. URL https://proceedings.mlr.press/v80/bai18a.html.

[6] Jye E. Beardow. Scroll, click, like, share, repeat: The algorithmic polarisation phenomenon. *ANU Journal of Law & Technology*, 2(1):153–164, 2021. Autumn 2021 issue.

[7] Omer Ben-Porat and Moshe Tennenholtz. Best response regression. *Advances in Neural Information Processing Systems*, 30, 2017.

[8] Omer Ben-Porat and Moshe Tennenholtz. Regression equilibrium. In *Proceedings of the 2019 ACM Conference on Economics and Computation*, pages 173–191, 2019.

[9] James Bennett and Stan Lanning. The Netflix prize. In *Proceedings of KDD Cup and Workshop*. ACM, 2007.

[10] Alain Berlinet and Christine Thomas-Agnan. *Reproducing kernel Hilbert spaces in probability and statistics*. Springer Science & Business Media, 2011.

[11] Andrew An Bian, Joachim M. Buhmann, Andreas Krause, and Sebastian Tschiatschek. Guarantees for greedy maximization of non-submodular functions with applications, 2019.

[12] Vivek S Borkar. *Stochastic approximation: a dynamical systems viewpoint*, volume 9. Springer, 2008.

[13] Avinandan Bose, Mihaela Curmei, Daniel L Jiang, Jamie Morgenstern, Sarah Dean, Lillian J Ratliff, and Maryam Fazel. Initializing services in interactive ml systems for diverse users. *arXiv preprint arXiv:2312.11846*, 2023.

[14] Jian-Feng Cai, Emmanuel J. Candès, and Zuowei Shen. A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, 20(4):1956–1982, 2010. URL https://doi.org/10.1137/080738970.

[15] Daniele Calandriello, Alessandro Lazaric, and Michal Valko. Second-order kernel online convex optimization with adaptive sketching. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 645–653. PMLR, 2017. URL https://proceedings.mlr.press/v70/calandriello17a.html.

[16] Daniele Calandriello, Luigi Carratino, Alessandro Lazaric, Michal Valko, and Lorenzo Rosasco. Gaussian process optimization with adaptive sketching: Scalable and no regret. *CoRR*, abs/1903.05594, 2019. URL http://arxiv.org/abs/1903.05594.

[17] Romain Camilleri, Kevin Jamieson, and Julian Katz-Samuels. High-dimensional experimental design and kernel bandits. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 1227–1237. PMLR, 2021. URL https://proceedings.mlr.press/v139/camilleri21a.html.

[18] Sarah H. Cen, Andrew Ilyas, Jennifer Allen, Hannah Li, and Aleksander Madry. Measuring strategization in recommendation: Users adapt their behavior to shape future content. In *ACM Conference on Economics and Computation (EC)*, 2024.

[19] Chandra Chekuri, Shalmoli Gupta, and Kent Quanrud. Streaming algorithms for submodular function maximization. In *Automata, Languages, and Programming: 42nd International Colloquium, ICALP 2015, Kyoto, Japan, July 6-10, 2015, Proceedings, Part I 42*, pages 318–330. Springer, 2015.

[20] Lin Chen, Andreas Krause, and Amin Karbasi. Interactive submodular bandit. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/file/f0935e4cd5920aa6c7c996a5ee53a70f-Paper.pdf.

[21] Ling Chen, Zhicheng Liu, Hong Chang, Donglei Du, and Xiaoyan Zhang. Online bp functions maximization. In *Algorithmic Aspects in Information and Management: 14th International Conference, AAIM 2020, Jinhua, China, August 10–12, 2020, Proceedings 14*, pages 173–184. Springer, 2020.

[22] Lixing Chen, Jie Xu, and Zhuo Lu. Contextual combinatorial multi-armed bandits with volatile arms and submodular reward. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper/2018/file/207f88018f72237565570f8a9e5ca240-Paper.pdf.

[23] Yeshwanth Cherapanamjeri, Constantinos Daskalakis, Andrew Ilyas, and Manolis Zampetakis. What makes a good fisherman? linear regression under self-selection bias. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing*, STOC 2023. Association for Computing Machinery, 2023. doi: 10.1145/3564246.3585177. URL https://doi.org/10.1145/3564246.3585177.

[24] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An open-source chatbot impressing GPT-4 with 90%* ChatGPT quality. https://lmsys.org/blog/2023-03-30-vicuna/, March 2023. Accessed: 2025.

[25] Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Hao Zhang, Banghua Zhu, Michael Jordan, Joseph E Gonzalez, and Ion Stoica. Chatbot arena: An open platform for evaluating LLMs by human preference. In *Proceedings of the 41st International Conference on Machine Learning*, 2024. URL https://arxiv.org/abs/2403.04132.

[26] Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Hao Zhang, Banghua Zhu, Michael Jordan, Joseph E. Gonzalez, and Ion Stoica. Chatbot arena: An open platform for evaluating LLMs by human preference. In *International Conference on Machine Learning (ICML)*, 2024.

[27] Federico Cinus, Marco Minici, Corrado Monti, and Francesco Bonchi. The effect of people recommenders on echo chambers and polarization. In *Proceedings of the Sixteenth International AAAI Conference on Web and Social Media (ICWSM '22)*, pages 90–101. Association for the Advancement of Artificial Intelligence (AAAI), 2022. ICWSM 2022.

[28] Michele Conforti and Gérard Cornuéjols. Submodular set functions, matroids and the greedy algorithm: tight worst-case bounds and some generalizations of the rado-edmonds theorem. *Discrete applied mathematics*, 7(3):251–274, 1984.

[29] Abhimanyu Das and David Kempe. Submodular meets spectral: Greedy algorithms for subset selection, sparse approximation and dictionary selection. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, ICML'11, page 1057–1064. Omnipress, 2011.

[30] Abhimanyu Das and David Kempe. Approximate submodularity and its applications: Subset selection, sparse approximation and dictionary selection. *Journal of Machine Learning Research*, 19(3):1–34, 2018. URL http://jmlr.org/papers/v19/16-534.html.

[31] Sarah Dean, Mihaela Curmei, Lillian Ratliff, Jamie Morgenstern, and Maryam Fazel. Emergent specialization from participation dynamics and multi-learner retraining. In *International Conference on Artificial Intelligence and Statistics*, pages 343–351. PMLR, 2024.

[32] Frances Ding, Moritz Hardt, John Miller, and Ludwig Schmidt. Retiring adult: New datasets for fair machine learning. *Advances in neural information processing systems*, 34:6478–6490, 2021.

[33] Robert Epstein and Ronald E. Robertson. The search engine manipulation effect (seme) and its possible impact on the outcomes of elections. *Proceedings of the National Academy of Sciences*, 112(33):E4512–E4521, 2015. URL https://www.pnas.org/doi/abs/10.1073/pnas.1419828112.

[34] Moran Feldman, Amin Karbasi, and Ehsan Kazemi. Do less, get more: Streaming submodular maximization with subsampling. *Advances in Neural Information Processing Systems*, 31, 2018.

[35] Sara Fish, Yannai A. Gonczarowski, and Ran I. Shorrer. Algorithmic collusion by large language models. *arXiv preprint arXiv:2404.00806*, 2024.

[36] Tony Ginart, Eva Zhang, Yongchan Kwon, and James Zou. Competing ai: How does competition feedback affect machine learning? In *International Conference on Artificial Intelligence and Statistics*, pages 1693–1701. PMLR, 2021.

[37] António Góis, Mehrnaz Mofakhami, Fernando P. Santos, Gauthier Gidel, and Simon Lacoste-Julien. Performative prediction on games and mechanism design. In *Proceedings of The 28th International Conference on Artificial Intelligence and Statistics*, volume 258 of *Proceedings of Machine Learning Research*, pages 1855–1863. PMLR, 2025.

[38] Daniel Golovin and Andreas Krause. Adaptive submodularity: Theory and applications in active learning and stochastic optimization. *J. Artif. Int. Res.*, 42(1):427–486, 2011.

[39] Peter M Guadagni and John DC Little. A logit model of brand choice calibrated on scanner data. *Marketing Science*, 2(3):203–238, 1983.

[40] Arnav Gudibande, Eric Wallace, Charlie Snell, Xinyang Geng, Hao Liu, Pieter Abbeel, Sergey Levine, and Dawn Song. The false promise of imitating proprietary LLMs. *arXiv preprint arXiv:2305.15717*, 2023.

[41] Carlos Guestrin, Andreas Krause, and Ajit Paul Singh. Near-optimal sensor placements in gaussian processes. In *Proceedings of the 22nd International Conference on Machine Learning*, ICML '05, page 265–272. Association for Computing Machinery, 2005. URL https://doi.org/10.1145/1102351.1102385.

[42] A. Guillory and J. Bilmes. Interactive submodular set cover. In *International Conference on Machine Learning (ICML)*, 2010.

[43] Lewis Hammond, Alan Chan, Jesse Clifton, Jason Hoelscher-Obermaier, Akbir Khan, Euan McLean, Chandler Smith, et al. Multi-agent risks from advanced AI. *Cooperative AI Foundation, Technical Report #1. arXiv preprint arXiv:2502.14143*, 2025.

[44] Moritz Hardt, Nimrod Megiddo, Christos Papadimitriou, and Mary Wootters. Strategic classification. In *Proceedings of the 2016 ACM Conference on Innovations in Theoretical Computer Science*, pages 111–122. ACM, 2016.

[45] F Maxwell Harper and Joseph A Konstan. The movielens datasets: History and context. *Acm transactions on interactive intelligent systems (tiis)*, 5(4):1–19, 2015.

[46] Keegan Harris, Chara Podimata, and Zhiwei Steven Wu. Strategic apple tasting. *Adv. Neural Inf. Process. Syst.*, abs/2306.06250, June 2023.

[47] Tatsunori Hashimoto, Megha Srivastava, Hongseok Namkoong, and Percy Liang. Fairness without demographics in repeated loss minimization. In *International Conference on Machine Learning*, pages 1929–1938. PMLR, 2018.

[48] Trevor Hastie, Robert Tibshirani, Jerome H Friedman, and Jerome H Friedman. *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer, 2009.

[49] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. URL https://arxiv.org/abs/1503.02531.

[50] Henning Hohnhold, Deirdre O'Brien, and Diane Tang. Focusing on the long-term: It's good for users and business. *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2015.

[51] Steven C. H. Hoi, Rong Jin, Jianke Zhu, and Michael R. Lyu. Batch mode active learning and its application to medical image classification. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, page 417–424. Association for Computing Machinery, 2006. URL https://doi.org/10.1145/1143844.1143897.

[52] Hong Huang, Bo Zhao, Hao Zhao, Zhou Zhuang, Zhenxuan Wang, Xiaoming Yao, Xinggang Wang, Hai Jin, and Xiaoming Fu. A cross-platform consumer behavior analysis of large-scale mobile shopping data. In *Proceedings of the 2018 World Wide Web Conference*, WWW '18, pages 1785–1794, Lyon, France, April 2018. International World Wide Web Conferences Steering Committee. doi: 10.1145/3178876.3186169. URL https://doi.org/10.1145/3178876.3186169.

[53] Ruben Interian, Ruslán G. Marzo, Isela Mendoza, and Celso C. Ribeiro. Network polarization, filter bubbles, and echo chambers: An annotated review of measures and reduction methods. *arXiv preprint*, 2022. arXiv:2207.13799.

[54] Roozbeh Irani-Kermani, Edward C. Jaenicke, and Ardalan Mirshani. Accommodating heterogeneity in brand loyalty estimation: Application to the U.S. beer retail market. *Journal of Marketing Analytics*, 11(4):820–835, 2023. doi: 10.1057/s41270-022-00187-2. URL https://doi.org/10.1057/s41270-022-00187-2.

[55] Meena Jagadeesan, Nikhil Garg, and Jacob Steinhardt. Supply-side equilibria in recommender systems. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[56] Meena Jagadeesan, Michael I Jordan, and Nika Haghtalab. Competition, alignment, and equilibria in digital marketplaces. *arXiv preprint arXiv:2208.14423*, 2023.

[57] Meena Jagadeesan, Michael I Jordan, Jacob Steinhardt, and Nika Haghtalab. Improved bayes risk can yield reduced social welfare under competition. *Advances in Neural Information Processing Systems*, 36, 2023.

[58] Hailey James, Chirag Nagpal, Katherine A Heller, and Berk Ustun. Participatory personalization in classification. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[59] Natasha Jaques, Angeliki Lazaridou, Edward Hughes, Caglar Gulcehre, Pedro A. Ortega, DJ Strouse, Joel Z. Leibo, and Nando de Freitas. Social influence as intrinsic motivation for multi-agent deep reinforcement learning. In *International Conference on Machine Learning (ICML)*, 2019.

[60] Julie Jiang, Xiang Ren, and Emilio Ferrara. Social media polarization and echo chambers in the context of covid-19: Case study. *JMIRx med*, 2(3):e29570, 2021. doi: 10.2196/29570.

[61] Jon Kleinberg, Sendhil Mullainathan, and Manish Raghavan. The challenge of understanding what users want: Inconsistent preferences and engagement optimization, 2022. URL https://arxiv.org/abs/2202.11776.

[62] Adam D. I. Kramer, Jamie E. Guillory, and Jeffrey T. Hancock. Experimental evidence of massive-scale emotional contagion through social networks. *Proceedings of the National*

Academy of Sciences, 111(24):8788–8790, 2014. URL https://www.pnas.org/doi/abs/10.1073/pnas.1320040111.

[63] Andreas Krause and Cheng Ong. Contextual gaussian process bandit optimization. In *Advances in Neural Information Processing Systems*, volume 24. Curran Associates, Inc., 2011. URL https://proceedings.neurips.cc/paper/2011/file/f3f1b7fc5a8779a9e618e1f23a7b7860-Paper.pdf.

[64] Branislav Kveton, Zheng Wen, Azin Ashkan, Hoda Eydgahi, and Brian Eriksson. Matroid bandits: Fast combinatorial optimization with learning. *CoRR*, abs/1403.5045, 2014. URL http://arxiv.org/abs/1403.5045.

[65] Yongchan Kwon, Antonio Ginart, and James Zou. Competition over data: how does data purchase affect users? *arXiv preprint arXiv:2201.10774*, 2022.

[66] Qiang Li, Chung-Yiu Yau, and Hoi-To Wai. Multi-agent performative prediction with greedy deployment and consensus seeking agents. In *Advances in Neural Information Processing Systems*, volume 35, pages 38449–38460, 2022.

[67] Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, Benjamin Newman, Binhang Yuan, Bobby Yan, Ce Zhang, Christian Cosgrove, Christopher D Manning, Christopher Ré, Diana Acosta-Navas, Drew A Hudson, Eric Zelikman, Esin Durmus, Faisal Ladhak, Frieda Rong, Hongyu Ren, Huaxiu Yao, Jue Wang, Keshav Santhanam, Laurel Orr, Lucia Zheng, Mert Yüksekgönül, Mirac Suzgun, Nathan Kim, Neel Guha, Niladri Chatterji, Omar Khessin, Peter Henderson, Qian Huang, Ryan Chi, Sang Michael Xie, Shibani Santurkar, Surya Ganguli, Tatsunori Hashimoto, Thomas Icard, Tianyi Zhang, Vishrav Chandu, William Wang, Xuechen Xie, Xuechen Zhang, Yushi Wang, Yuntao Zhou, and Yuta Koreeda. Holistic evaluation of language models. *Transactions on Machine Learning Research*, 2023. URL https://arxiv.org/abs/2211.09110.

[68] Yancheng Liang, Daphne Chen, Abhishek Gupta, Simon S. Du, and Natasha Jaques. Learning to cooperate with humans using generative agents. *arXiv preprint arXiv:2411.13934*, 2024.

[69] Zhicheng Liu, Ling Chen, Hong Chang, Donglei Du, and Xiaoyan Zhang. Online algorithms for bp functions maximization. *Theoretical Computer Science*, 858:114–121, 2021.

[70] Zhicheng Liu, Longkun Guo, Donglei Du, Dachuan Xu, and Xiaoyan Zhang. Maximization problems of balancing submodular relevance and supermodular diversity. *Journal of Global Optimization*, 82(1):179–194, 2022. URL https://doi.org/10.1007/s10898-021-01063-6.

[71] Jérémie Mary, Romaric Gaudel, and Philippe Preux. Bandits and recommender systems. In *Machine Learning, Optimization, and Big Data: First International Workshop, MOD 2015, Taormina, Sicily, Italy, July 21-23, 2015, Revised Selected Papers 1*, pages 325–336. Springer, 2015.

[72] Eric Mazumdar, Lillian J. Ratliff, and S. Shankar Sastry. On gradient-based learning in continuous games. *SIAM Journal on Mathematics of Data Science*, 2(1):103–131, 2020. doi: 10.1137/18M1231298. URL https://doi.org/10.1137/18M1231298.

[73] John Miller, Juan C Perdomo, and Tijana Zrnic. Outside the echo chamber: Optimizing the performative risk. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 7710–7720. PMLR, 2021.

[74] Sumeet Motwani, Wolfram Barfuss, Lewis Hammond, Caspar Oesterheld, and Vincent Conitzer. Secret collusion among AI agents: Multi-agent deception via steganography. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

[75] Adhyyan Narang, Evan Faulkner, Dmitriy Drusvyatskiy, Maryam Fazel, and Lillian J. Ratliff. Multiplayer performative prediction: Learning in decision-dependent games. *Journal of Machine Learning Research*, 24(202):1–56, 2023.

[76] George L Nemhauser and Laurence A Wolsey. Best algorithms for approximating the maximum of a submodular set function. *Mathematics of operations research*, 3(3):177–188, 1978.

[77] George L Nemhauser, Laurence A Wolsey, and Marshall L Fisher. An analysis of approximations for maximizing submodular set functions—i. *Mathematical programming*, 14(1):265–294, 1978.

[78] Randal S Olson, William La Cava, Patryk Orzechowski, Ryan J Urbanowicz, and Jason H Moore. PMLB: A large benchmark suite for machine learning evaluation and comparison. *BioData Mining*, 10(1):36, 2017.

[79] Isaac Ong, Amjad Almahairi, Vincent Wu, Wei-Lin Chiang, Tianhao Wu, Joseph E. Gonzalez, M. Waleed Kadous, and Ion Stoica. RouteLLM: Learning to route LLMs with preference data. In *International Conference on Learning Representations (ICLR)*, 2025.

[80] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744, 2022.

[81] Orestis Papadigenopoulos and Constantine Caramanis. Recurrent submodular welfare and matroid blocking semi-bandits. In *Advances in Neural Information Processing Systems*, volume 34, pages 23334–23346. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper/2021/file/c44bebb973e14fe539676e0e9155b121-Paper.pdf.

[82] Juan Perdomo, Tijana Zrnic, Celestine Mendler-Dünner, and Moritz Hardt. Performative prediction. In *International Conference on Machine Learning*, pages 7599–7609. PMLR, 2020.

[83] Georgios Piliouras and Fang-Yi Yu. Multi-agent performative prediction: From global stability and optimality to chaos. In *Proceedings of the 24th ACM Conference on Economics and Computation*, pages 1047–1048. ACM, 2023.

[84] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C Berg, and Li Fei-Fei. ImageNet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

[85] Omid Sadeghi and Maryam Fazel. Online continuous DR-submodular maximization with long-term budget constraints. In *Proc. International conference on Artificial Intelligence and Statistics*, pages 4410–4419, 2020.

[86] Omid Sadeghi and Maryam Fazel. Differentially private monotone submodular maximization under matroid and knapsack constraints. In *International Conference on Artificial Intelligence and Statistics*, pages 2908–2916. PMLR, 2021.

[87] Omid Sadeghi and Maryam Fazel. Fast first-order methods for monotone strongly DR-submodular maximization. *arXiv preprint arXiv:2111.07990*, 2021.

[88] Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[89] Eric M Schwartz, Eric T Bradlow, and Peter S Fader. Customer acquisition via display advertising using multi-armed bandit experiments. *Marketing Science*, 36(4):500–522, 2017.

[90] H.J. Scudder. Probability of error of some adaptive pattern-recognition machines. *IEEE Transactions on Information Theory*, 11(3):363–371, July 1965.

[91] Matthias W Seeger, Sham M Kakade, and Dean P Foster. Information consistency of nonparametric gaussian process methods. *IEEE Transactions on Information Theory*, 54(5):2376–2382, 2008.

[92] Pier Giuseppe Sessa, Maryam Kamgarpour, and Andreas Krause. Bounding inefficiency of equilibria in continuous actions games using submodularity and curvature. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 2017–2027. PMLR, 2019.

[93] Burr Settles. Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison, 2009.

[94] Eliot Shekhtman and Sarah Dean. Strategic usage in a multi-learner setting. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 2024.

[95] Ilia Shumailov, Zakhar Shumaylov, Yiren Zhao, Nicolas Papernot, Ross Anderson, and Yarin Gal. Ai models collapse when trained on recursively generated data. *Nature*, 631(8022):755–759, 2024.

[96] Niranjan Srinivas, Andreas Krause, Sham Kakade, and Matthias Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design. In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, ICML'10, page 1015–1022. Omnipress, 2010.

[97] Peter Steinberger and contributors. OpenClaw: Open-source agentic AI framework, 2025. URL https://github.com/openclaw/openclaw. GitHub repository.

[98] Jinyan Su and Sarah Dean. Learning from streaming data when users choose. *arXiv [cs.LG]*, June 2024.

[99] Maxim Sviridenko, Jan Vondrák, and Justin Ward. Optimal approximation for submodular and supermodular optimization with bounded curvature. *Mathematics of Operations Research*, 42(4):1197–1218, 2017.

[100] Sho Takemori, Masahiro Sato, Takashi Sonoda, Janmajay Singh, and Tomoko Ohkuma. Submodular bandit problem under multiple constraints. In *Proceedings of the 36th Conference on Uncertainty in Artificial Intelligence (UAI)*, volume 124 of *Proceedings of Machine Learning Research*, pages 191–200. PMLR, 2020. URL https://proceedings.mlr.press/v124/takemori20a.html.

[101] Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Stanford Alpaca: An instruction-following LLaMA model. https://github.com/tatsu-lab/stanford_alpaca, 2023. Accessed: 2025.

[102] Michal Valko, Nathaniel Korda, Rémi Munos, Ilias N. Flaounas, and Nello Cristianini. Finite-time analysis of kernelised contextual bandits. *ArXiv*, abs/1309.6869, 2013.

[103] Johannes von Oswald, Eyvind Niklasson, Ettore Randazzo, João Sacramento, Alexander Mordvintsev, Andrey Zhmoginov, and Max Vladymyrov. Transformers learn in-context by gradient descent. In *International Conference on Machine Learning (ICML)*, 2023.

[104] Guanghui Wang, Ioannis Panageas, Georgios Piliouras, and Fang-Yi Yu. Last-iterate convergence for symmetric, general-sum, 2×2 games under the exponential weights dynamic. *arXiv preprint arXiv:2502.08063*, 2025.

[105] Xiaolu Wang, Chung-Yiu Yau, and Hoi To Wai. Network effects in performative prediction games. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 36514–36540. PMLR, 2023.

[106] Kai Wei, Rishabh Iyer, and Jeff Bilmes. Submodularity in data subset selection and active learning. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 1954–1963. PMLR, 2015. URL https://proceedings.mlr.press/v37/wei15.html.

[107] John Werner. Did deepseek copy off of openai? and what is distillation? *Forbes*, 2025. URL https://www.forbes.com/sites/johnwerner/2025/01/30/did-deepseek-copy-off-of-openai-and-what-is-distillation/.

[108] Mark Wilhelm, Ajith Ramanathan, Alexander Bonomo, Sagar Jain, Ed H. Chi, and Jennifer Gillenwater. Practical diversified recommendations on youtube with determinantal point processes. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, CIKM '18, page 2165–2173. Association for Computing Machinery, 2018. URL https://doi.org/10.1145/3269206.3272018.

[109] Xiaohan Xu, Ming Li, Chongyang Tao, Tao Shen, Reynold Cheng, Jinyang Li, Can Xu, Dacheng Tao, and Tianyi Zhou. A survey on knowledge distillation of large language models. *arXiv preprint arXiv:2402.13116*, 2024.

[110] Yue Yang, Yongbin Sun, Zhengrui Cao, Yun Zhang, Jiaqiang Li, and Yanhao Liu. A survey on recommendation ecosystem: Addressing the multi-stakeholder perspective. *arXiv preprint arXiv:2402.15046*, 2024.

[111] Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations (ICLR)*, 2023.

[112] David Yarowsky. Unsupervised word sense disambiguation rivaling supervised methods. In *33rd Annual Meeting of the Association for Computational Linguistics*, pages 189–196, Cambridge, Massachusetts, USA, June 1995. Association for Computational Linguistics. doi: 10.3115/981658.981684.

[113] Houssam Zenati, Alberto Bietti, Eustache Diemert, Julien Mairal, Matthieu Martin, and Pierre Gaillard. Efficient kernelized ucb for contextual bandits. In *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pages 5689–5720. PMLR, 2022. URL https://proceedings.mlr.press/v151/zenati22a.html.

[114] Tong Zhang. Effective dimension and generalization of kernel learning. In *Advances in Neural Information Processing Systems*, volume 15. MIT Press, 2002. URL https://proceedings.neurips.cc/paper_files/paper/2002/file/25db67c5657914454081c6a18e93d6dd-Paper.pdf.

[115] Xueru Zhang, Mohammadmahdi Khaliligarekani, Cem Tekin, et al. Group retention when using machine learning in sequential decision making: the interplay between user dynamics and fairness. *Advances in Neural Information Processing Systems*, 32, 2019.

[116] Ying Zhang, Tao Xiang, Timothy M Hospedales, and Huchuan Lu. Deep mutual learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4320–4328, 2018.

[117] Zaiwei Zhu, Runyu Wan, Yingbin Cho, Haoming Luo, Zhuoran Yang, and Zhaoran Wang. Online performative gradient descent for learning nash equilibria in decision-dependent games. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

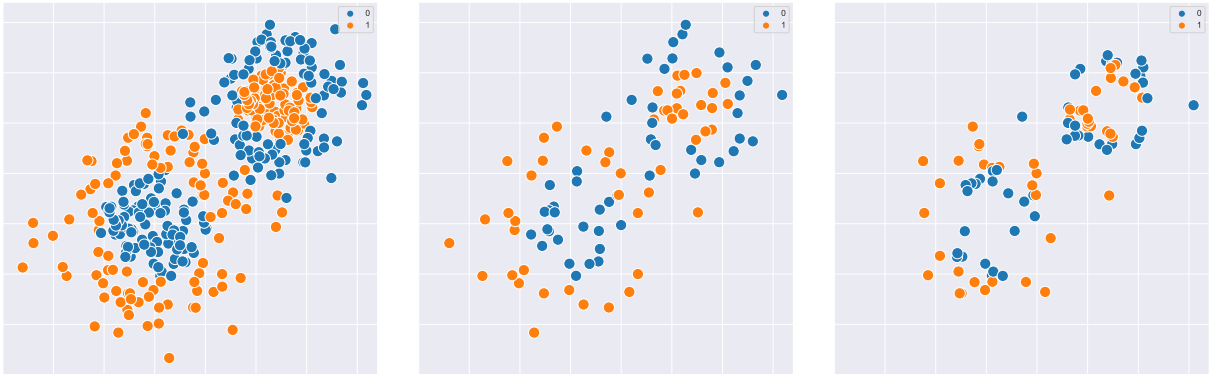
Appendix A

Appendix for Chapter 2

A.1 Table of Notation

Notation	Description
V	Ground set of items
m	Number of set functions
h_q	q-th set function, $q \in [m]$
u_t	Index of arrived function at time t
ϕ_{u_t}	Context vector for function h_{u_t} at time t
v_t	Item selected at time t
$S_{k,q}$	Items selected for function h_q up to time k
y_t	Noisy marginal gain feedback at time t
$y_{f,t}, y_{g,t}$	Separate submodular and supermodular feedback at time t
S_q^*	Optimal set for function h_q
T_q	Number of items selected for function h_q by time T
κ_f, κ^g	Submodular and supermodular curvatures
γ, ζ	Submodularity ratio and generalized curvature
$\mathcal{R}_{\mathrm{BP}}(T), \mathcal{R}_{\mathrm{WS}}(T)$	Regret for BP and WS functions
$\mathcal{R}_{\mathrm{BP},2}(T)$	Regret for BP functions with separate feedback
$\Delta(\phi, S, v)$	Marginal gain of adding v to S for context ϕ
$\mathcal{K}$	Reproducing kernel Hilbert space (RKHS)
B	Bound on RKHS norm of Δ
G_t	Nyström set at time t
β_t	Exploration-exploitation tradeoff parameter
$d_{\mathrm{eff}}(\lambda, T)$	Effective dimension
l_1, l_2	Modular lower bounds for f and g
f_1, g_1	Totally normalized f and g
$\pi_j(v\|A)$	Distorted marginal gain for selecting v given A at step j

Table A.1: Table of key notation used in this chapter.



(a) Original training set (b) Submodular objective (c) BP objective

Figure A.1: Greedy algorithm selection on submodular (second panel) and BP (third panel) objectives for subset selection of 100 points of training data from a ground set of 400 points. The first panel depicts the entire training (ground) set. The details are provided in Section A.6.

A.2 Applications and Role of MNN functions

In this section, we present two examples that illustrate the modeling power of BP functions for different applications and compare this approach with common approaches from the literature.

A.2.1 Active Learning

From Figure A.1, we see that the BP function (third panel) results in the selection of complementary points near the decision boundary—i.e., points of opposite class that are proximal. It is *impossible* to choose a submodular function that encourages this type of desirable cooperative behavior due to the diminishing-returns property.

Comparison with approaches for Pool-based Active Learning In their survey paper, [93] compare submodularity-based approaches with other approaches for active learning. The main benefit of framing the active learning problem as submodular is that the greedy algorithm can be employed, which is much less computationally expensive than other common active learning approaches. While submodularity has been shown to be relevant to active learning [41, 51, 106], [93] remark that in general, the active learning problem cannot be framed as submodular.

In our paper, by extending the classes of functions that can be optimized online, we take a step towards addressing this limitation of submodularity. Further, an open question outlined in [93] is that of multi-task active learning, which has not been explored extensively in previous work.

However, our formulation in Vignette 2.2 naturally extends to this multi-task setting.

A.2.2 Recommendation Systems

In Table A.2, the BP function enables the desirable selection of movies from the same series in the correct order. As above, it is impossible to design a submodular utility function that encourages this type of behavior.

Comparison with approaches for online recommendation For recommender systems, the dependencies between past and future recommendations may be modeled through a changing "state variable," leading to adopting reinforcement learning (RL) solutions [2]. These have been tremendously effective at maximizing engagement; however, [61] highlight that a key oversight of these approaches is that the click and scroll-time data that platforms observe is not representative of the users' actual utilities: "research has demonstrated that we often make choices in the moment that are inconsistent with what we actually want." Hence, Kleinberg et al. advocate to encode diminishing returns of addictive but superficial content into the model, in the manner that we do with submodular functions. Further, RL systems are incentivized to manipulate users' behavior [50, 108], mood [62] and preferences [33]; this inspires the use of principled mathematical techniques, as in the present work, to design systems to behave as we want rather than simply following the trail of the unreliable observed data.

	SM Objective	BP Objective
0	Lion King, The	Godfather, The
1	Speed	Godfather: Part II, The
2	Godfather, The	Godfather: Part III, The
3	Godfather: Part II, The	Star Wars: Episode I
4	Terminator, The	Memento
5	Good Will Hunting	Harry Potter I
6	Memento	Star Wars: Episode II
7	Harry Potter I	Harry Potter: II
8	Dark Knight, The	Star Wars: Episode III
9	Inception	Dark Knight, The

Table A.2: Comparison of the selections of the greedy algorithm on submodular and BP objectives for movie recommendation, on a toy ground set of 23 movies from the MovieLens dataset. The submodular objective is the facility location objective, chosen from [20]. In the BP objective, there is an additional reward at each step for choosing a movie that is complementary with previously selected movies; this results the desirable joint **selection of groups of movies from the same series**. The task is formalized mathematically in Section 3.5, and experimental details are provided in the supplement.

A.3 A simple approach to guarantee low regret: Why it is too weak

In this section, we provide an alternate proof for the approximation ratio that the greedy algorithm obtains on a BP function in the offline setting. The robustness of this proof can be very simply studied, in a manner similar to [20]. However, the approximation ratio obtained is worse than that of [5]; hence, the regret guarantee in the online setting would be provided against a weak baseline. This motivates why we revisit the proof from [5]. Just for this section, we use simpler notation h for the BP function and k for the cardinality constraint, since we are presenting the argument for the offline setting.

Proposition A.1. *For a BP maximization problem subject to a cardinality constraint,* $\max_{S:|S|\leq k} h(S)$ *where* $h(S) = f(S) + g(S)$*, the greedy algorithm obtains the following guarantee:*

$$h(S) \geq (1 - e^{-(1-\kappa^g)})h(S^*),$$

where $S^* = \{v_1^*, \ldots, v_k^*\} = arg\max_{S:|S|\leq k} h(S)$ *and* κ^g *is the curvature of the supermodular function* g*.*

Lion King, The	Good Will Hunting
Speed	Godfather: Part III, The
True Lies	Star Wars: Episode I - The Phantom Menace
Aladdin	Gladiator
Dances with Wolves	Memento
Batman	Shrek
Godfather, The	Harry Potter I: The Sorcerer's Stone
Godfather: Part II, The	Star Wars: Episode II - Attack of the Clones
Terminator, The	Harry Potter II: The Chamber of Secrets
Indiana Jones and the Last Crusade	Star Wars: Episode III - Revenge of the Sith
Men in Black	Dark Knight, The
	Inception

Table A.3: Ground set for Table A.2

Proof. For $t < k$, let $S_t = \{v_1, \ldots, v_t\}$ be the items chosen by the greedy algorithm. We can write:

$$\begin{aligned}
h(S^*) &\leq h(S^* \cup S_t) \\
&= h(S_t) + \sum_{j=1}^{k} h(v_j^* | S_t \cup \{v_1^*, \ldots, v_{j-1}^*\}) \\
&\leq h(S_t) + \frac{1}{1-\kappa^g} \sum_{j=1}^{k} h(v_j^* | S_t) \\
&\leq h(S_t) + \frac{1}{1-\kappa^g} \sum_{j=1}^{k} h(v_{t+1} | S_t) \\
&= h(S_t) + \frac{k}{1-\kappa^g} \big(h(S_{t+1}) - h(S_t)\big),
\end{aligned}$$

where the first inequality uses Lemma C.1.(ii) of [5] and the second inequality is due to the update rule of the greedy algorithm. Rearranging the terms, we can write:

$$\begin{aligned}
h(S^*) - h(S_t) &\leq \frac{k}{1-\kappa^g} \big([h(S^*) - h(S_t)] - [h(S^*) - h(S_{t+1})]\big) \\
h(S^*) - h(S_{t+1}) &\leq (1 - \frac{1-\kappa^g}{k})(h(S^*) - h(S_t))
\end{aligned}$$

Applying the above inequality recursively for $t = 0, \ldots, k-1$, we have:

$$h(S^*) - h(S) \leq (1 - \frac{1-\kappa^g}{k})^k (h(S^*) - \underbrace{h(\emptyset)}_{=0})$$

Using the inequality $1 - x \leq e^{-x}$ and rearranging the terms, we have:

$$h(S) \geq (1 - e^{-(1-\kappa^g)})h(S^*)$$

If $\kappa_f = 1$, this approximation ratio matches the obtained approximation ratio for the greedy algorithm in Theorem 3.7 of [5] without the need to change the original proof of the greedy algorithm. □

A.4 Proofs from Section 2.4

A.4.1 Approximate Greedy on BP Functions

This section contains the supporting technical lemmas for the proofs of Lemmas 2.1 and 2.2 (whose main proofs appear in Section 2.4.1).

Notation We use S_t to refer to the ordered set of elements chosen for function h until round t, and S to refer to the ordered final set of items chosen for function h until round T. Hence, S_j refers to the first j elements chosen for h. Let s_j be the j^{th} element of S. Then, we define $a_j = h(s_j|\{s_1 \dots s_{j-1}\})$ be the gain of the j^{th} element chosen.

Recall that S is an ordered set. We let $C \subseteq [k]$ denote the indices (in increasing order) of elements in S that are also in S^*. For instance, for $S = \{s_1 \dots s_5\}$ and $S \cap S^* = \{s_1, s_2, s_3\}$, we have $C = \{1, 2, 3\}$. Hence, $j \in C \iff s_j \in S \cap S^*$. Further, define filtered sets $C_t = \{c \in C | c \leq t\}$ as the subset of the first t_{th} elements of S that are also in the optimal S^*.

The lemma below is a modified version of Equation (19) in [5], which accounts for the deviation of our algorithm from the greedy policy.

Lemma A.2. *Using the notation above and for S as chosen by the approximate greedy procedure, it follows that $\forall t \in [k]$,*

$$h(S^*) \leq \kappa_f \sum_{j \in [t-1] \setminus C_{t-1}} (a_j + r_j) + \sum_{j \in C_{t-1}} (a_j + r_j) + \frac{k - |C_{t-1}|}{1 - \kappa^g} (a_t + r_t)$$

Proof of Lemma A.2. By the properties of BP functions from Lemma C.2 in [5], it follows for all

$t \in [k]$ that

$$h(S^*) \leq \kappa_f \sum_{j \in [t-1] \setminus C} a_j + \sum_{j \in C_{t-1}} a_j + h(S^* \setminus S_{t-1} | S_{t-1}) \tag{A.1}$$

$$\leq \kappa_f \sum_{j \in [t-1] \setminus C} (a_j + r_j) + \sum_{j \in C} (a_j + r_j) + h(S^* \setminus S_{t-1} | S_{t-1}) \tag{A.2}$$

The inequality above follows because the coefficients on the first two summations are positive and $r_j \geq 0$. Now, we must simplify the third term to obtain the desired. For any feasible v,

$$h(v|S_{t-1}) \leq \sup_v h(v|S_{t-1}) \leq h(s_t|S_{t-1}) + r_t. \tag{A.3}$$

The first inequality follows from the definition of sup and the second follows from the definition of r_t in the proof of Theorem 2.4 above. Now, apply inequality (iv) from Lemma C.1 in [5]:

$$\begin{aligned} h(S^* \setminus S_{t-1} | S_{t-1}) &\leq \frac{1}{1 - \kappa^g} \sum_{v \in S^* \setminus S_{t-1}} h(v|S_{t-1}) \\ &\leq \frac{1}{1 - \kappa^g} \sum_{v \in S^* \setminus S_{t-1}} h(s_t|S_{t-1}) + r_t \end{aligned}$$

The second line follows from Equation (A.3).

We have that

$$|S^* \setminus S_{t-1}| = |S^*| - |S^* \cap S_{t-1}| = k - |S^* \cap S_{t-1}|.$$

Hence,

$$h(S^* \setminus S_{t-1} | S_{t-1}) \leq \frac{k - |S^* \cap S_{t-1}|}{1 - \kappa^g} \left[h(s_t|S_{t-1}) + r_t \right]$$

Recognizing that $|S^* \cap S_{t-1}| = |C_{t-1}|$ completes the argument. □

A.4.2 Approximate Greedy on WS Functions

Define S, s_j, a_j, C as in the proof for BP functions. The proof of Lemma 2.2 appears in Section 2.4.1. Below, we present the supporting technical lemma for the WS case, analogous to Lemma A.2. The proof for Lemma A.3 is different than Lemma A.2 due to the change in the class of functions being

considered. The similarity of the two proofs suggests the generality of our proof technique and indicates that it may be analogously applied to other classes of functions as well.

The lemma below is a modified version of Lemma 1 in [11].

Lemma A.3. *Using the notation above and for S as chosen by the approximate greedy procedure, it follows that $\forall t \in \{0 \dots k-1\}$,*

$$h(S^*) \leq \zeta \sum_{j \in [t] \setminus C_t} (a_j + r_j) + \sum_{j \in C_t} (a_j + r_j) + \frac{1}{\gamma}(k - |C_t|)\,(a_{t+1} + r_{t+1})$$

Proof of Lemma A.3. The proof follows from the definitions of generalized curvature, submodularity ratio, and instantaneous regret r_t.

$$h(S^* \cup S_t) = h(S^*) + \sum_{j \in [t]} h(s_j | S^* \cup S_{j-1})$$

We can split the summation above to separately consider the elements from S_t that do and do not overlap with S^*.

$$\begin{aligned} h(S^* \cup S_t) &= h(S^*) + \sum_{j : s_j \in S_t \setminus S^*} h(s_j | S^* \cup S_{j-1}) + \underbrace{\sum_{j : s_j \in S_t \cap S^*} h(s_j | S^* \cup S_{j-1})}_{=0} \\ &= h(S^*) + \sum_{j : s_j \in S_t \setminus S^*} h(s_j | S^* \cup S_{j-1}) \end{aligned} \tag{A.4}$$

From the definition of submodularity ratio,

$$h(S^* \cup S_t) \leq h(S^*) + \frac{1}{\gamma} \sum_{\omega \in S^* \setminus S_t} h(\omega | S_t) \tag{A.5}$$

From the definition of generalized curvature, it follows that

$$\begin{aligned} \sum_{j : s_j \in S_t \setminus S^*} h(s_j | S^* \cup S_{j-1}) &\geq (1 - \zeta) \sum_{j : s_j \in S_t \setminus S^*} h(s_j | S_{j-1}) \\ &= (1 - \zeta) \sum_{j : s_j \in S_t \setminus S^*} a_{j+1} \end{aligned} \tag{A.6}$$

Then, plugging the inequalities (A.5) and (A.6) into (A.4),

$$h(S^*) = h(S^* \cup S_t) - \sum_{j:s_j \in S_t \setminus S^*} h(s_j | S^* \cup S_{j-1})$$
$$\leq \left[h(S) + \frac{1}{\gamma} \sum_{\omega \in S^* \setminus S} h(\omega | S) \right] + \left[\zeta \sum_{j:s_j \in S_t \setminus S^*} a_{j+1} - \sum_{j:s_j \in S_t \setminus S^*} a_{j+1} \right] \tag{A.7}$$

Now, we can rearrange and write

$$h(S) - \sum_{j:s_j \in S_t \setminus S^*} a_{j+1} = \sum_{j:s_j \in S_t \cap S^*} a_{j+1}$$

to simplify Equation (A.7) as

$$h(S^*) = \zeta \sum_{j:s_j \in S_t \setminus S^*} a_{j+1} + \frac{1}{\gamma} \sum_{\omega \in S^* \setminus S} h(\omega | S) + \sum_{j:s_j \in S_t \cap S^*} a_{j+1}$$
$$\leq \zeta \sum_{j:s_j \in S_t \setminus S^*} a_{j+1} + \frac{1}{\gamma} \sum_{\omega \in S^* \setminus S} (a_{t+1} + r_t) + \sum_{j:s_j \in S_t \cap S^*} a_{j+1} \tag{A.8}$$
$$\leq \zeta \sum_{j:s_j \in S_t \setminus S^*} (a_{j+1} + r_j) + \sum_{j:s_j \in S_t \cap S^*} (a_{j+1} + r_j) + \frac{1}{\gamma}(k - |C_t|)(a_{t+1} + r_t) \tag{A.9}$$

Equation (A.8) follows by using the definitions of r_t and supremum, and Equation (A.9) follows since $r_t \geq 0$. □

A.4.3 Approximate Weighted Greedy on BP Functions

We recap notation for ease of reference. Define the modular lower bound of the submodular function $l_1(S) = \sum_{j \in S} f(j | V \setminus \{j\})$. Additionally, define the totally normalized submodular function as $f_1(S) = f(S) - l_1(S)$. Note that the f_1 will always have curvature $\kappa_f = 1$. $h(S) = f_1(S) + g(S) + l_1(S)$. Now, define the function

$$\pi_j(v|A) = \left(1 - \frac{1}{k}\right)^{k-j-1} f_1(v|A) + g(v|A) + l_1(v) \tag{A.10}$$

Also define

$$\pi_j(A) = \left(1 - \frac{1}{k}\right)^{k-j} f_1(A) + g(A) + l_1(A) \tag{A.11}$$

The proof of Lemma 2.3 appears in Section 2.4.2. Below, we present the supporting technical lemmas used in that proof.

Lemma A.4.

$$\pi_j(s_j|S_{j-1}) + r_j \geq \frac{1}{k}\left(1 - \frac{1}{k}\right)^{k-(j+1)} (f(S^*) - f(S_j) + \frac{1}{k} l(S^*)$$

Proof of A.4. From the definition of r_j,

$$\begin{aligned}
\pi_j(s_j|S_{j-1}) + r_j &\geq \frac{1}{k} \sum_{e \in S^*} \pi_j(e|S_{j-1}) \\
&= \frac{1}{k} \sum_{e \in S^*} \left(1 - \frac{1}{k}\right)^{k-(j+1)} f_1(e|S_{j-1}) + g_1(e|S_{j-1}) + l(e) \\
&\geq \frac{1}{k}\left(1 - \frac{1}{k}\right)^{k-(j+1)} (f_1(S^*) - f_1(S_{j-1})) + \frac{1}{k} l(S^*)
\end{aligned}$$

The inequality follows from the submodularity of f_1 and the supermodular curvature of g_1.

□

Lemma A.5. *Any approximately weighted greedy procedure with constants $\{r_j\}_{j=1}^k$ returns a set S of size k such that*

$$f_1(S) + g_1(S) + l(S) + \sum_{j=1}^{k} r_j \geq \left(1 - \frac{1}{e}\right) f_1(S^*) + l(S^*)$$

Proof. According to the definition of π, we have that $\pi_0(\emptyset) = 0$ and

$$\pi_k(S) = f_1(S) + g_1(S) + l(S)$$

Applying Lemma 4 from [70], we have

$$
\begin{aligned}
&\pi_{j+1}(S_{j+1}) - \pi_j(S_j) \\
&= \pi_j(s_{j+1}|S_j) + \frac{1}{k}\left(1-\frac{1}{k}\right)^{k-(j+1)} f_1(S_j) \\
&\geq \frac{1}{k}\left(1-\frac{1}{k}\right)^{k-(j+1)} f_1(S^*) + \frac{1}{k} l(S^*) - r_{j+1}
\end{aligned}
$$

Above, we applied Lemma A.4 to obtain the inequality. Now, we have that

$$
\begin{aligned}
f_1(S) + g_1(S) + l(S) &= \sum_{j=0}^{k-1} \pi_{j+1}(S_{j+1}) - \pi_j(S_j) \\
&\geq \sum_{j=0}^{k-1} \frac{1}{k}\left(1-\frac{1}{k}\right)^{k-(j+1)} f_1(S^*) + \frac{1}{k} l(S^*) - r_{j+1} \\
&\geq \left(1-\frac{1}{e}\right) f_1(S^*) + l(S^*) - \sum_{j=1}^{k} r_j
\end{aligned}
$$

□

A.5 Discussion and Proofs from Section 2.5 and Section 2.6

A.5.1 Remarks on hyperparameters (η, b)

Note that b refers to our budget fraction variable as it serves to limit the final size of G_t, while η is an accuracy-computation tradeoff variable that tends to produce larger G_t's. While η and b are somewhat related (and are partially redundant) we utilize the "budget" and "accuracy" notion as originally defined in [113] to be consistent with that work

A.5.2 Remarks on step size β_t

From on the analysis found in Zenati et al. [113], we set

$$
\beta_t = \sqrt{\lambda} B + \sqrt{4\log(T) + \log\left(e + \frac{et}{\lambda}\right) d_{\text{eff}}} \tag{A.12}
$$

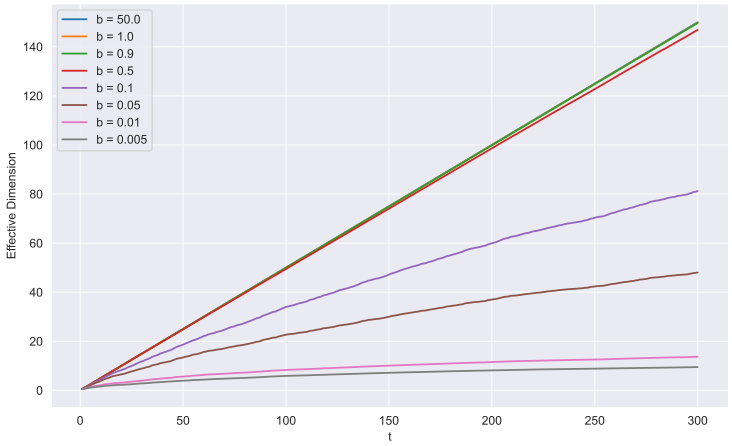


Figure A.2: The dependence of effective dimension d_{eff} as on the parameter b in the RBF kernel.

which enables our regret bounds to hold where $e = \exp(1)$, λ is a hyperparameter, and B is our RKHS norm bound.

In our empirical simulations, however, we found it much more effective to set β_t to a constant which is then tuned as a hyperparameter. In fact, Zenati et al. [113] found this to be the case in their simulations as well.

A.5.3 Remark on role of kernel parameters on d_{eff}

Consider the RBF kernel $\hbar(x, x') = \exp(-b\|x - x'\|^2)$. If the parameter b is very large, then the kernel function will be very close to zero for all $x \neq x'$. Hence, the kernel matrix K_T will be close to the identity matrix, and the eigenvalues will decay very slowly. Hence the effective dimension d_{eff} is likely to be large. Our current regret bound does not capture this, because we wanted to focus on the scaling of regret with T. However, there is a constant in front that scales as b, which effectively changes the base of the $\log(T)$ in the regret bound [91, Section 4.B]. In Figure A.2, we see that if the horizon T is quite small, this effect can dominate and make the T-scaling appear almost linear. On the other hand, if we make b very small, then the quantity B would increase; this is because $\hbar(x, x')$ being large is not very informative about the function values at x and x'. Hence, some care is required to tune the kernel parameters correctly. This applies to other kernel functions as well. This effect is present in prior works [63, 96, 113] as well, but these do not address it explicitly which is why we wanted to offer some clarity about this point.

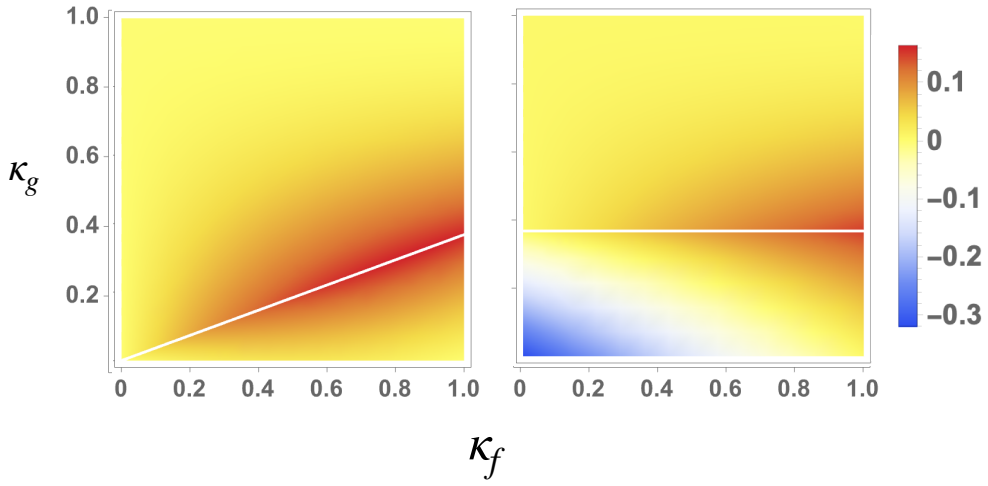


Figure A.3: Contour plot of (left) $F_1(\kappa_{f,q}, \kappa_q^g) = \min\left\{1 - \frac{\kappa_{f,q}}{e},\ 1 - \kappa_q^g\right\} - \frac{1}{\kappa_{q,f}}\left[1 - e^{-(1-\kappa_q^g)\kappa_{q,f}}\right]$ and (right) $F_2(\kappa_{f,q}, \kappa_q^g) = \min\left\{1 - \frac{1}{e},\ 1 - \kappa_q^g\right\} - \frac{1}{\kappa_{q,f}}\left[1 - e^{-(1-\kappa_q^g)\kappa_{q,f}}\right]$. (Left) compares the α from Theorem 2.5 with that from Theorem 2.4, and (right) compares α from Proposition A.6 with that from Theorem 2.4.

A.5.4 Proof of Theorem 2.4(b)

The proof appears in Section 2.5. The main contribution lies in showing Lemma A.3 (above); this shows that the offline counterpart, Lemma 1 from [11], holds in the online setting as well.

A.5.5 Remarks on [70]

Comparison of α for greedy vs weighted Greedy. In Figure A.3, we compare the α of the greedy optimization of the BP function in [5] with the distorted greedy variant in Equation (2.15). In the left panel, we see that the α in Equation (2.15) is everywhere greater.

Error in [70] and proposed fix. In [70] on the bottom of Pg.188, the authors use the inequality:

$$\sum_{e \in \text{OPT}} g_1(e|S_t) \geq (1 - \kappa^g)(g_1(OPT) - g_1(S_t))$$

Consider the following counterexample with $|V| = 3$ and $k = 2$ as the cardinality constraint. Define $g(S) = |S|^2$, which is a concave over modular function, so it is supermodular. We can verify from definitions that $\kappa^g = 0.8$ and the modular lower bound $l_2(S) = |S|$, so that $g_1(S) = |S|^2 - |S|$. For simplicity, consider the case where $t = 0$, so that $S_t = \emptyset$. Then, plugging into the equation, we see that the LHS is 0, whereas the RHS is $0.2 \times 4 = 0.8 > 0$. Hence, this is a contradiction. This

example can be easily generalized to any concave over modular function, larger ground set sizes or different t.

We rectify this by swapping this inequality with

$$g_1(e|S_t) \geq (1 - \kappa^{g_1})(g_1(\text{OPT}) - g_1(S_t)) = 0$$

The equality above holds because $\kappa^{g_1} = 1$ by construction. Hence, the g_1 term disappears from the analysis. The proof of Theorem 2.5, which incorporates this fix, appears in Section 2.6.2.

A.5.6 Analysis without Assumption (2a)

In certain applications, Assumption (2a) on $l_{q,1}$ may not be reasonable. For these cases, we may modify the algorithm slightly, and provide an alternative bound, that is slightly weaker. Consider a modified version of Algorithm 4, where line 3 is substituted with:

$$\text{Set } y_t = (1 - 1/T_{u_t})^{T_{u_t} - (t_{u_t}+1)} \, y_{f,t} + y_{g,t} \tag{A.13}$$

Recognize that this Algorithm does not require Assumption (2a).

Define

$$\mathcal{R}_{\text{BP, 3}}(T) := \sum_{q=1}^{m} \min\left\{1 - \frac{1}{e},\ 1 - \kappa_q^g\right\} h_q(S_q^*) - h_q(S_q). \tag{A.14}$$

Observe from the right panel of Figure A.3 that the α in the definition above is still better than that of [5] for most choices of $\kappa_{f,q}, \kappa_q^g$. Now, we can state our modified result. The proof follows similarly to Theorem 2.5.

Proposition A.6. *Let Assumption 2.1 and Assumption (2b) and Assumption (2c) hold. Additionally, let the conditions on ϵ_t hold as in Theorem 2.5. Then Algorithm 4 with the modification above yields* $\mathbb{E}[\mathcal{R}_{BP,\ 3}(T)] \leq O\left(\sqrt{T}\left(B\sqrt{\lambda d_{eff}} + d_{eff}\right)\right)$

Proof of Proposition A.6. For the modified version of the algorithm described in equation (A.13), the analysis is almost identical. Repeating the analysis of Lemma A.5 and Lemma A.4, we obtain:

$$f_q(S_q) + g_{q,1}(S_q) + l_{q,2}(S_q) + \sum_{j=1}^{T_q} r_{m_j} \geq \left(1 - \frac{1}{e}\right) f_q(S_q^*) + l_{q,2}(S_q^*)$$

Then, we can follow the same arguments as in Lemma 2.3 to conclude:

$$f_q(S) + g_q(S) \geq \min\left\{1 - \frac{1}{e},\ 1 - \kappa_q^g\right\}\ h_q(S_q^*) - \sum_{j=1}^{T_q} r_{m_j}.$$

□

A.5.7 Remarks on Guess-and-double technique to replace Assumption (2c)

In this section, we provide a heuristic argument for why we expect that guess-and-double techniques should not affect the overall regret scaling in Theorem 2.5.

In the traditional multi-armed bandit, when the time horizon T is unknown, the proposed method of dealing with this is to start with an initial guess $\widehat{T} = 1$ and then double each time the current time step crosses our latest guess. Any parameters in the algorithm that depend on T (step size for e.g) are set based on $\widehat{T}$ instead. This divides the entire horizon into phases, one for each guess $\widehat{T}$. Then, for each phase, the regret must be sublinear because this is equivalent to playing a shorter game with known horizon. Since the regret is the accumulation of the regrets of each phase, the overall regret must be sublinear as well.

However, in our case, the situation is more intricate because the overall regret is not expressible as the summation of regret over phases. Hence, the original style of argument does not apply. What we do then, is to keep track of the change in regret due to setting the distortion co-efficient in terms of $\widehat{T_q}$ instead of T_q. We choose $\widehat{T_q} = \min\{2^j : j > t\}$.

When T_q is known, the distortion $D_t = \left(1 - \frac{1}{T_{u_t}}\right)^{T_{u_t} - t_{u_t} - 1}$ increases monotonically from $\left(1 - \frac{1}{T_{u_t}}\right)^{T_{u_t}-1}$ to 1 with t_{u_t} i.e as more elements are added. This monotonicity is used in the original argument to obtain the sublinear regret guarantee.

However, when the guess-and-double technique is used, the distortion is no longer monotonic in t_{u_t}. Within each phase, D_t increases from $\left(1 - \frac{1}{\widehat{T_{u_t}}}\right)^{\widehat{T_{u_t}}-1}$ to 1 but then reduces once $\widehat{T_{u_t}}$ is updated at the end of the phase. It turns out that the regret actually decreases within the phase (compared to the situation where we know T_q) due to the increased distortion, but increases in the transitions between the phases. Below, we characterize the changes in regret in the two cases.

Define

$$\widehat{\Lambda}_{j,q}(x, A) = \left(1 - \frac{1}{\widehat{T_q}}\right)^{\widehat{T_q}-(j+1)} f_{q,1}(x|A) + g_{q,1}(x|A) + l_q(x)$$

Analogously, we can define

$$\widehat{\pi}_{j,q}(S) = \left(1 - \frac{1}{\widehat{T_q}}\right)^{\widehat{T_q}-j} f_{q,1}(S) + g_{q,1}(S) + l_q(S)$$

Case 1: Within phase Previously Lemma 4 from [70], we had

$$\begin{aligned}
&\pi_{j+1,q}(S_{j+1,q}) - \pi_{j,q}(S_{j,q}) \\
&= \Lambda_{j,q}(s_j, S_{j,q}) + \frac{1}{T_q}\left(1 - \frac{1}{T_q}\right)^{T_q-(j+1)} f_{q,1}(S_{j,q})
\end{aligned}$$

Now, we can replace this conclusion with

$$\begin{aligned}
&\widehat{\pi}_{j+1,q}(S_{j+1,q}) - \widehat{\pi}_{j,q}(S_{j,q}) \\
&= \widehat{\Lambda}_{j,q}(s_j, S_{j,q}) + \frac{1}{\widehat{T_q}}\left(1 - \frac{1}{\widehat{T_q}}\right)^{\widehat{T_q}-(j+1)} f_{q,1}(S_{j,q}) \\
&= \widehat{\Lambda}_{j,q}(s_j, S_{j,q}) + \frac{1}{\widehat{T_q}}\left(1 - \frac{1}{\widehat{T_q}}\right)^{\widehat{T_q}-(j+1)} f_{q,1}(S_{j,q}) + \underbrace{\left(\frac{1}{\widehat{T_q}} - \frac{1}{T_q}\right)\left(1 - \frac{1}{\widehat{T_q}}\right)^{\widehat{T_q}-(j+1)} f_1(S_{j,q})}_{N_{\text{within},j}}
\end{aligned}$$

The term $N_{\text{within},j}$ is a new term. The remainder of the proof goes through as expected, while these additional terms propagate through the proof.

Case 2: Between phase Note that if step j is in a different phase than step $j+1$, it follows that the distortion at step j is

$$\left(1 - \frac{1}{\widehat{T_q}}\right)^{\widehat{T_q}-\widehat{T_q}} = 1.$$

Since step $t+1$ is the first time step in a phase, it follows that the guess for $\widehat{T_q}$ just doubled, and is $\widehat{T_q} = 2i$. Then, the distortion for step $j+1$ is

$$\left(1 - \frac{1}{2i}\right)^{t-1}$$

As in the Case 1, we can track the extra term from Lemma 4, which in this case is

$$N_{\text{between},j} = -\left(1 - \left(1 - \frac{1}{2i}\right)^{t-1}\right) f_1(S_{j,q})$$

As before, this new term propagates through the proof.

Putting it together Accounting for the new terms, our modified final statement of Lemma 2.3

$$h_q(S_q) \geq \min\left\{1 - \frac{\kappa_{f,q}}{e},\ 1 - \kappa_q^g\right\} h_q(S_q^*) - \sum_{j=1}^{T_q} r_{m_j} + \sum_{j:\text{change}} N_{\text{between},j} + \sum_{j:\text{no change}} N_{\text{within},j}$$

Above the indices $(j : \text{change})$ include the $\log(T_q)$ time steps, which are the first time step in a phase i.e the first time step after our guess $\widehat{T_q}$ was recently updated; the indices $(j : \text{no change})$ include all other time steps. Hence, the new term

$$N = \sum_{j:\text{change}} N_{\text{between},j} + \sum_{j:\text{no change}} N_{\text{within},j}$$

gets subtracted from the regret. We observe that each of the $N_{\text{between},j}$ terms are positive and there are many of these: $T_q - \log(T_q)$ to be precise. However, the $N_{\text{within},j}$ terms are negative and increase the regret; however, there are only $\log(T_q)$ of these. While it is difficult to quantify the terms exactly, there is no strong reason to believe that the few negative terms greatly outweigh the positive terms. From preliminary simulations, we find that the regret remains roughly the same with the doubling trick; we leave an extensive experimental investigation of this to future work.

A.6 Details on Experiments

Details for Table A.3, Table A.2 The chosen toy ground set of 23 elements is detailed in Table A.3. The submodular function is the facility location function; we chose this function because it is used in prior work [20] for the task of movie recommendation. The supermodular part is the sum-sum-dispersion function, and the weights that capture the complementarity between movies are specified in the python notebook `code/table-1.ipynb` in the attached code.

From Table A.2, we notice that with the submodular objective, the greedy algorithm chooses the

first two movies in the Godfather series but does not choose the third. Similarly, it chooses the first Harry Potter but not the subsequent ones. In contrast, with the BP function, the greedy algorithm chooses all elements from the series in both cases. This behavior cannot be encoded using solely a submodular function, but it is very easy to do so with a BP function.

Setup for movie recommendation in Figure 2.1 From MovieLens and using the matrix-completion approach in [14], we obtain a ratings matrix $M \in \mathbb{R}^{900\times 1600}$, where $M_{i,j}$ is the rating of the i_{th} user for the j^{th} movie; for density of data, we consider the most active users and most popular movies.

We cluster the users into $m = 10$ groups using the k-means algorithm and design a BP objective for each user-group. The objective for the q_{th} group is decomposed as $h_q(A) = \sum_{v\in A} m_q(v) + \lambda_1 f_q(A) + \lambda_2 g_q(A)$, where the modular part $m_q(v)$ is the average rating for movie v amongst all users in group k.

Let the set L refer to the collection of all genres in the ground set. The concave-over-modular submodular part encourages the recommender to maintain a balance across genres in chosen suggestions: $f_q(A) = \sum_{g\in L} \sqrt{1 + u_{q,g}(A)}$. The set L is the collection of all genres. We now specify what $u_{q,g}(\cdot)$ is. For each element $v \in V$, define a vector $r(v) \in \{0,1\}^{|L|}$. Here, each entry corresponds to a genre and is 1 if the genre is associated with the movie v. Then let $N_v = r(v)^\top \mathbf{1}$ denote the number of genres for movie v. In $f_q(\cdot)$, we specify

$$u_{q,g}(A) = \sum_{v\in A} \mathbf{1}(m_q(v) > \tau) \frac{\mathbf{1}(\text{v has genre g})}{N_v}$$

Above, $\mathbf{1}$ is the indicator function.

The supermodular function, in contrast is designed to encourage the optimizer to exploit complementarities within genres $g_q(A) = \sum_{g\in L} \left(1 + \tilde{u}_{q,g}(A)\right)^2$, where we define

$$\tilde{u}_{q,g}(A) = \sum_{v\in A} \mathbf{1}(\text{v has genre g}(m_k(v) > \tau)) \frac{m_q(v)}{N_v}$$

We want the complementarities to be amplified when the movies have higher ratings, so notice that each term in $\tilde{u}_{q,g}$ is scaled by $m_q(v)$ relative to each term of $u_{q,g}$.

The constants λ_1, λ_2 were chosen such that the supermodular part slightly dominates the

submodular part, since previous works already study functions that are primarily submodular. The code is contained in notebook "Figure 2."

Kernel Estimation for Figure 2.1 For Algorithm 1, we choose the RBF kernel for movies, the linear kernel for users and the Jaccard kernel for a history of recommendations. The composite kernel $\hbar((u,v,A),(u',v',A')) = \kappa_1\hbar_{\text{user}}(u,u') + \kappa_2\hbar_{\text{movie}}(v,v') + \kappa_3\hbar_{\text{history}}(A,A')$ for $\kappa_1,\kappa_2,\kappa_3 > 0$. For Algorithm 4, we choose the RBF kernel for o_t.

Active Learning. This corresponds to Vignette 2.2 with $m = 1$ tasks. We apply the Naive-Bayes formulation of active learning in Equation (5) of [106] and set the submodular part as $f(A) = f^{\text{NB}}(A)$. The supermodular part is the sum-sum-dispersion function as above $g(A) = \sum_{v_t \in A} \sum_{v_j \in A: v_j \neq v_t} B_{t,j}$. Here $B_{t,j} = 0$ if (v_t, v_j) are from the same class, and $B_{t,j} = 1/\text{dist}(v_t, v_j)$ if (v_t, v_j) are from the opposite class; this encourages the selection of proximal points from different classes.

Here, we elaborate on the choice of submodular function. Assume our features are discrete - each point $v \in V$ has features $x_v \in \mathcal{X}$ (where $\mathcal{X}$ is some finite set) and binary label $y_v \in \{0,1\}$, denoted by the orange and blue colors in Figure A.1. Then, for any $(x \in \mathcal{X}, y \in \{0,1\})$ and for any subset of training points $S \subseteq V$, we can define

$$m_{x,y}(S) = \sum_{v \in S} \mathbf{1}(x_v = x \wedge y_v = y)$$

as the empirical count of the joint occurrence of (x,y) in S. Then, inspired by the construction in Wei et al., we define the submodular part f as

$$f(S) = \sum_{x \in \mathcal{X}} \sum_{y \in \{0,1\}} \sqrt{m_{x,y}(V)} \log(m_{x,y}(S))$$

To obtain the finite set $\mathcal{X}$, we discretize our 2-dimensional features into 56 boxes. The square-root in the expression above does not occur in the original paper and was introduced by us due to better empirical performance. The intuition for constructing $f(\cdot)$ in this way is that the feature x should appear alongside label y in the chosen subset with roughly the same frequency as in the ground training set.

Appendix B

Appendix for Chapter 3

B.1 Terminology and Preliminary Results

B.1.1 Dynamical systems terminology

This subsection records standard definitions underlying terms such as "invariant set" and "internally chain transitive" in Lemma B.6. Let $\dot{x}(t) = h(x(t))$ be an ODE with locally Lipschitz h, so that solutions are unique. Let $\phi_t(x)$ denote the state at time $t \geq 0$ of the solution initialized at x at time 0.

Definition B.1 (Invariant set)**.** *A set $A \subseteq \mathbb{R}^d$ is (forward)* invariant *for the ODE if $\phi_t(x) \in A$ for all $x \in A$ and all $t \geq 0$.*

Definition B.2 ((ε, T)-chain)**.** *Fix $\varepsilon > 0$ and $T > 0$. An* (ε, T)-chain in A from x to y *is a finite sequence of points $x = x_0, x_1, \ldots, x_k = y$ in A and times $t_1, \ldots, t_k \geq T$ such that*

$$\|\phi_{t_i}(x_{i-1}) - x_i\| < \varepsilon \qquad \text{for all } i \in \{1, \ldots, k\}.$$

Definition B.3 (Internally chain transitive)**.** *A compact invariant set A is* internally chain transitive *if for every $x, y \in A$ and every $\varepsilon > 0$, $T > 0$, there exists an (ε, T)-chain in A from x to y.*

B.1.2 Regularity of Squared and Cross Entropy Losses

Lemma B.1 (Regularity of the squared-error loss)**.** *Let $\mathcal{X} \subset \mathbb{R}^d$, $\mathcal{Y} \subset \mathbb{R}$, $W \subset \mathbb{R}^d$ be compact. Write*

$$\ell(x, y; \theta) \;=\; \big(y - x^\top \theta\big)^2, \qquad B = \max_{\theta \in W} \|\theta\|.$$

Then for all $(x, y) \in \mathcal{X} \times \mathcal{Y}$ and all $\theta_1, \theta_2 \in W$:

(i) ℓ is nonnegative, C^∞ in θ, convex, and β–smooth with $\beta = 2R^2$.

(ii) ℓ is locally Lipschitz on W:

$$\big|\ell(x, y; \theta_1) - \ell(x, y; \theta_2)\big| \le L_W \, \|\theta_1 - \theta_2\|, \quad L_W = R\big(2|y| + 2RB\big).$$

(iii) The Hessian is constant in θ, so $\|\nabla^2 \ell(\theta_1) - \nabla^2 \ell(\theta_2)\| = 0$ (i.e. $\gamma_\ell = 0$).

Proof. (i) One computes $\nabla_\theta \ell = -2\, x\big(y - x^\top \theta\big)$ and $\nabla^2_\theta \ell = 2\, x\, x^\top \succeq 0$. Hence $\ell \ge 0$, is C^∞, convex, and $\|\nabla \ell(\theta_1) - \nabla \ell(\theta_2)\| \le 2R^2 \|\theta_1 - \theta_2\|$.

(ii) Observe

$$\ell(\theta_1) - \ell(\theta_2) = \big(y - x^\top \theta_1\big)^2 - \big(y - x^\top \theta_2\big)^2 = x^\top(\theta_2 - \theta_1)\, \big(2y - x^\top(\theta_1 + \theta_2)\big).$$

Since $\|\theta_i\| \le B$, $|x^\top(\theta_2 - \theta_1)| \le R\|\theta_1 - \theta_2\|$ and $\big|2y - x^\top(\theta_1 + \theta_2)\big| \le 2|y| + 2RB$, giving the stated bound.

(iii) Because $\nabla^2_\theta \ell \equiv 2\, x\, x^\top$ is independent of θ, its difference vanishes.

□

Lemma B.2. *Under Assumption 3.1, for all $z \in \mathcal{Z}$ the loss $\ell(z, \cdot)$ is non-negative, convex, differentiable, locally Lipschitz and β_ℓ-smooth with*

$$\beta_\ell = 2R^2 \quad \textit{for both the squared loss and cross-entropy loss.}$$

Moreover, let

$$\mathcal{R}(\theta) = \mathbb{E}_{z \sim \mathcal{P}}[\ell(z, \theta)], \quad \theta^* = \arg\min_\theta f(\theta), \quad \epsilon = f(\theta^*).$$

Then for any θ with $\|\theta - \theta^*\| \le \gamma$, the following quadratic upper bound holds:

$$\mathcal{R}(\theta) \;\le\; \epsilon + \tfrac{\beta_\ell}{2}\,\|\theta - \theta^*\|^2 \;\le\; \epsilon + R^2\,\gamma^2.$$

Proof. The first part (non-negativity, convexity, differentiability, local Lipschitzness, and smoothness) follows directly by applying Lemmas B.1 and B.3 under Assumption 3.1, which together show each $\ell(z, \theta)$ is β_ℓ-smooth with $\beta_\ell = 2R^2$.

Since $f(\theta) = \mathbb{E}_z[\ell(z, \theta)]$, differentiability and smoothness carry over to f, and we have

$$\|\nabla f(\theta) - \nabla f(\theta^*)\| \le \beta_\ell\,\|\theta - \theta^*\|.$$

At the minimizer θ^*, $\nabla f(\theta^*) = 0$. Hence for any θ with $\|\theta - \theta^*\| \le \gamma$, the gradient satisfies

$$\|\nabla f(\theta)\| \le \beta_\ell\,\|\theta - \theta^*\| \le \beta_\ell\,\gamma.$$

Applying the standard smoothness inequality (second-order Taylor upper bound) around θ^*,

$$f(\theta) \le f(\theta^*) + \langle \nabla f(\theta^*),\, \theta - \theta^* \rangle + \tfrac{\beta_\ell}{2}\,\|\theta - \theta^*\|^2 = \epsilon + \tfrac{\beta_\ell}{2}\,\|\theta - \theta^*\|^2,$$

where we used $\nabla f(\theta^*) = 0$ and $f(\theta^*) = \epsilon$. Finally, substituting $\beta_\ell = 2R^2$ gives

$$f(\theta) \le \epsilon + R^2\,\|\theta - \theta^*\|^2 \le \epsilon + R^2\,\gamma^2,$$

completing the proof. □

Lemma B.3 (Regularity of the multiclass cross-entropy loss)**.** *Let $\mathcal{X} \subset \mathbb{R}^d$, $\mathcal{Y} = \{1, \dots, K\}$, $W \subset \mathbb{R}^{K \times d}$ compact, and let $\|x\| \le R$ for all $x \in \mathcal{X}$. Define for $(x, y) \in \mathcal{X} \times \mathcal{Y}$ and $\Theta = (\theta_1, \dots, \theta_K) \in W$*

$$\ell(x, y; \Theta) \;=\; -\log \frac{\exp(\theta_y^\top x)}{\sum_{k=1}^K \exp(\theta_k^\top x)}.$$

Then for all x, y, Θ_1, Θ_2 as above:

(i) ℓ is nonnegative, C^∞ in Θ, convex, and β–smooth with $\beta = R^2$.

(ii) ℓ is (locally) Lipschitz on W:

$$\big|\ell(x,y;\Theta_1)-\ell(x,y;\Theta_2)\big| \;\le\; L\,\|\Theta_1-\Theta_2\|, \qquad L=\sqrt{2}\,R.$$

Proof. The proofs of each subpart are as follows:

(i) Let $p=\mathrm{softmax}(\Theta\,x)\in\Delta^{K-1}$. One checks

$$\nabla_\Theta \ell=(p-e_y)\,x^\top, \qquad \nabla^2_\Theta \ell=\big[\mathrm{diag}(p)-p\,p^\top\big]\;\otimes\;\big(x\,x^\top\big).$$

Since $\mathrm{diag}(p)-p\,p^\top\succeq 0$ $\ell\ge 0$, is C^∞ and convex, and

$$\|\nabla^2_\Theta \ell\| \;\le\; \|x\|^2\,\lambda_{\max}\big(\mathrm{diag}(p)-p\,p^\top\big) \;\le\; R^2,$$

it follows that ℓ is β–smooth with $\beta=R^2$.

(ii) By the mean-value theorem,

$$\big|\ell(\Theta_1)-\ell(\Theta_2)\big| \;\le\; \sup_{\Theta\in W}\|\nabla_\Theta\ell\|\;\|\Theta_1-\Theta_2\|.$$

But

$$\|\nabla_\Theta\ell\|=\|p-e_y\|\;\|x\| \;\le\; \sqrt{2}\,R,$$

giving the claimed Lipschitz constant $L=\sqrt{2}\,R$.

□

Lemma B.4. *Let $\{h_\theta\}_{\theta\in\Theta}$ be the family of softmax classifiers mapping $x\in\mathbb{R}^d$ to a distribution over C classes,*

$$h_\theta(x)_c=\frac{e^{x^\top\theta_c}}{\sum_{j=1}^{C}e^{x^\top\theta_j}}.$$

Assume:

(i) $\|x\|_2\le R$ for all x in the support of $\mathscr{P}$.

(ii) There exists $\alpha>0$ such that for all x,θ,c, we have $h_\theta(x)_c\ge\alpha$.

(iii) θ, θ^* *lie in a convex set* $\Theta \subset \mathbb{R}^{C\times d}$.

Then the mapping $\theta \mapsto h_\theta(x)$ *is Lipschitz in the* ℓ_1*–norm: for all* $\theta, \theta^* \in \Theta$,

$$\|h_\theta(x) - h_{\theta^*}(x)\|_1 \;\le\; L_h \,\|\theta - \theta^*\|_2, \quad L_h = R\sqrt{2\,(1-\alpha)}.$$

Proof. By the multivariate mean-value theorem, there exists $\bar\theta$ on the line segment between θ^* and θ such that

$$h_\theta(x) - h_{\theta^*}(x) = D_\theta h(\bar\theta; x)\,\big(\theta - \theta^*\big),$$

where $D_\theta h(\bar\theta; x) \in \mathbb{R}^{C\times(Cd)}$ is the Jacobian w.r.t. all parameters. Write its cth row as the gradient vector $\nabla_\theta h_c(\bar\theta; x)$. A direct calculation shows for each class c and each block k:

$$\nabla_{\theta_k} h_c(\bar\theta; x) = h_c\big(\mathbf{1}\{c=k\} - h_k\big)\,x \quad\Longrightarrow\quad \|\nabla_\theta h_c(\bar\theta; x)\|_2 \le \|x\|_2\, h_c \sqrt{(1-h_c)^2 + \sum_{k\neq c} h_k^2}.$$

Using $\sum_k h_k = 1$ and the lower-bound $h_c \ge \alpha$, one checks

$$(1-h_c)^2 + \sum_{k\neq c} h_k^2 = (1-h_c)(1+h_c) \le 2\,(1-h_c) \le 2\,(1-\alpha).$$

Hence

$$\|\nabla_\theta h_c(\bar\theta; x)\|_2 \le R\, h_c\, \sqrt{2\,(1-\alpha)}.$$

Therefore the operator norm from ℓ_2 to ℓ_1 satisfies

$$\|D_\theta h(\bar\theta; x)\|_{2\to 1} \;=\; \sum_{c=1}^{C} \|\nabla_\theta h_c(\bar\theta; x)\|_2 \;\le\; R\sqrt{2(1-\alpha)} \sum_{c=1}^{C} h_c = R\sqrt{2(1-\alpha)}.$$

Putting these together gives

$$\|h_\theta(x) - h_{\theta^*}(x)\|_1 \le \|D_\theta h(\bar\theta; x)\|_{2\to 1}\, \|\theta - \theta^*\|_2 \le L_h\, \|\theta - \theta^*\|_2,$$

as claimed. □

Lemma B.5. *Under Assumption 3.1, for all* $z \in \mathcal{Z}$, *the loss* $\ell(z, \cdot)$ *is non-negative, convex, differentiable, locally Lipschitz, and* β_ℓ*-smooth.*

Proof. The result follows by applying Lemmas B.1 and B.3 with $|x| \le R$ and choosing $\beta_\ell = \max 2R^2, R^2 = 2R^2$. Each loss is nonnegative, convex, differentiable, locally Lipschitz, and smooth under these bounds. □

B.2 Convergence of Algorithm 5 and Algorithm 6

B.2.1 Convergence of Algorithm 5

The convergence results for MSGD (Algorithm 5) follow as special cases of the MSGD-P analysis when the probing set $U = \emptyset$. In this case, all probing terms vanish and the augmented potential $\widetilde{f}$ reduces to the original potential f.

Lemma 3.1. *Let Assumptions 3.1–3.4 hold. Then the iterates $\{\Theta^t\}$ of Algorithm 5 converge to a compact connected internally chain transitive invariant set of the ODE $\dot{\Theta} = -\nabla f(\Theta)$.*

Proof. This follows from the proof of Theorem 3.6 with $U = \emptyset$. When $U = \emptyset$:

- The indicator $\mathbf{1}_{i \in U} = 0$ for all $i \in [m]$, so all probing terms vanish.
- The augmented potential reduces to $\widetilde{f}(\Theta) = f(\Theta)$.
- The ODE (3.16) becomes $\dot{\Theta} = -\nabla f(\Theta)$.

The stochastic approximation argument proceeds identically: by Lemma B.6, verifying local Lipschitzness of $-\nabla f$ (Lemma B.5, Lemma B.9), the step-size conditions (Assumption 3.2), the martingale variance bound (Lemma B.7 with $U = \emptyset$, so $K_{\text{probe}} = 0$), and bounded iterates (Assumption 3.4), the iterates converge almost surely to a compact connected internally chain transitive invariant set of $\dot{\Theta} = -\nabla f(\Theta)$. □

Theorem 3.2. *Let Assumptions 3.1–3.4 hold. Then the iterates $\{\Theta^t\}$ of Algorithm 5 converge to the set of stationary points $\{\Theta : \nabla f(\Theta) = 0\}$ almost surely.*

Proof. Follows directly from Theorem 3.6 with $U = \emptyset$. □

Algorithm 7 Multi-learner Streaming Gradient Descent

Require: loss function $\ell(\cdot,\cdot) \geq 0$; Initial models $\Theta^0 = (\theta_1^0, \cdots, \theta_k^0)$; Learning rate $\{\eta^t\}_{t=1}^{T+1}$

1: **for** $t = 0, 1, 2, \ldots, T$ **do**
2: Sample data point $z^t \sim \mathscr{P}$
3: User selects model $i = M(z^t; \widetilde{\Theta}^t)$
4: $\theta_i^{t+1} \leftarrow \theta_i^t - \eta^t \nabla \ell(z^t, \theta_i^t)$
5: **end for**
6: **return** Θ^T

B.2.2 Convergence of MSGD-P (Algorithm 6)

Define the empirical probing loss and augmented potential:

$$\hat{L}_i(\theta_i) = \tfrac{1}{n} \sum_{q=1}^{n} \ell(\tilde{z}_i^q, \theta_i), \quad i \in U, \tag{B.1}$$

$$\widetilde{f}(\Theta) = f(\Theta) + p \sum_{i \in U} \left(\hat{L}_i(\theta_i) + \frac{\lambda}{2} \|\theta_i\|^2 \right). \tag{B.2}$$

Define the ordinary differential equation (ODE) $\dot{\Theta} = \widetilde{F}(\Theta)$ with:

$$\widetilde{F}_i(\Theta) = -\Big(\tau \alpha_i \, \mathbb{E}_{z \sim \mathscr{P}_i}[\nabla \ell(z, \theta_i)] + (1-\tau) a_i(\Theta) \, \mathbb{E}_{z \sim \mathscr{D}_i(\Theta)}[\nabla \ell(z, \theta_i)] + p \, \mathbf{1}_{i \in U} \left(\nabla \hat{L}_i(\theta_i) + \lambda \, \theta_i \right) \Big) \tag{B.3}$$

Lemma B.6 (Borkar [12], Chapter 2, Theorem 2)**.** *Let $\{x_n\}$ be a sequence generated by the stochastic approximation algorithm*

$$x_{n+1} = x_n + a(n)[h(x_n) + M_{n+1}], \quad n \geq 0,$$

where:

1. *$h : \mathbb{R}^d \to \mathbb{R}^d$ is Lipschitz continuous*
2. *The step sizes $\{a(n)\}$ satisfy $\sum_n a(n) = \infty$ and $\sum_n a(n)^2 < \infty$*
3. *$\{M_n\}$ is a martingale difference sequence satisfying $E[||M_{n+1}||^2 | \mathcal{F}_n] \leq K(1 + ||x_n||^2)$ for some $K > 0$*
4. *$\sup_n ||x_n|| < \infty$ almost surely*

Then almost surely, the sequence $\{x_n\}$ converges to a (possibly sample path dependent) compact

connected internally chain transitive invariant set of the ODE

$$\dot{x}(t) = h(x(t)).$$

Lemma B.7 (Martingale variance bound)**.** *Suppose that Assumption 3.1 holds and that the probing datasets $\{\mathfrak{D}_i\}_{i\in U}$ are fixed finite sets. Let $g^t(\Theta^t) = (g_1^t(\Theta^t), \dots, g_m^t(\Theta^t))$ be the stochastic update vector defined by*

$$g_i^t(\Theta^t) \;=\; g_{i,\mathrm{plat}}^t(\Theta^t) \;+\; g_{i,\mathrm{probe}}^t(\Theta^t),$$

where

$$g_{i,\mathrm{plat}}^t(\Theta^t) = \begin{cases} \nabla\ell(z^t, \theta_i^t), & \text{if } i = i_t, \\ 0, & \text{if } i \neq i_t, \end{cases} \qquad g_{i,\mathrm{probe}}^t(\Theta^t) = p\,\mathbf{1}_{i\in U}\left(\nabla\ell(\tilde{z}_i^t, \theta_i^t) + \lambda\,\theta_i^t\right),$$

with $\tilde{z}_i^t$ sampled uniformly from $\mathfrak{D}_i$ independently of i_t. Define the filtration $\mathcal{F}_t = \sigma(\Theta^0, z^1, \dots, z^{t-1}, \{\mathfrak{D}_i\}_{i\in U})$, which contains all information available prior to time t. Define the martingale difference sequence:

$$v^t = g^t(\Theta^t) - \mathbb{E}[g^t(\Theta^t) \mid \mathcal{F}_t].$$

Then there exists a constant $K > 0$ (made explicit below) such that:

$$\mathbb{E}[\|v^t\|^2 \mid \mathcal{F}_t] \le K\,(1 + \|\Theta^t\|^2).$$

Proof. Gradient bounds. From Lemmas B.1 and B.3, for all (x, y) with $\|x\| \le R$ and all θ we have bounds of the form

$$\|\nabla_\theta \ell(x, y; \theta)\| \le A_0 + A_1\,\|\theta\|,$$

with

$$(A_0, A_1) = \begin{cases} (2R\,Y_{\max},\ 2R^2), & \text{(squared loss)}, \\ (\sqrt{2}\,R,\ 0), & \text{(cross-entropy)}. \end{cases}$$

For the probe term under squared loss, labels are pseudo-labels $\hat{y} = \mathrm{median}\{\langle x, \theta_0^j\rangle : j \in [m]\}$ so that $|\hat{y}| \le R\,\max_{j\in[m]}\|\theta_0^j\| =: Y_{\max,\mathrm{probe}}$, giving $\|\nabla_\theta\ell(x, \hat{y}; \theta)\| \le \widetilde{A}_0 + A_1\,\|\theta\|$ with $\widetilde{A}_0 := 2R\,Y_{\max,\mathrm{probe}}$.

Including the L2 regularization term (which appears only in the probe update) and using the triangle inequality,

$$\|p\,(\nabla_\theta \ell(x,\hat{y};\theta)+\lambda\,\theta)\| \le p\,\widetilde{A}_0 + p\,(A_1+\lambda)\,\|\theta\|.$$

Define the block constants

$$K_{\text{plat}} := 2\max\big(A_0^2, A_1^2\big), \qquad K_{\text{probe}} := 2\max\big((p\,\widetilde{A}_0)^2, \big(p\,(A_1+\lambda)\big)^2\big),$$

where for cross-entropy we take $A_1 = 0$ and use the same $A_0 = \sqrt{2}R$ for both platform and probe.

Variance decomposition. By definition of v^t and $\mathrm{Var}(Y) = \mathbb{E}[\|Y\|^2] - \|\,\mathbb{E}[Y]\|^2$,

$$\mathbb{E}[\|v^t\|^2 \mid \mathscr{F}_t] \;\le\; \mathbb{E}[\|g^t(\Theta^t)\|^2 \mid \mathscr{F}_t].$$

We bound the RHS. Since g^t has at most one nonzero platform block and up to $|U|$ nonzero probe blocks,

$$\|g^t(\Theta^t)\|^2 = \sum_{i=1}^m \|g^t_{i,\text{plat}} + g^t_{i,\text{probe}}\|^2 \;\le\; 2\sum_{i=1}^m \|g^t_{i,\text{plat}}\|^2 + 2\sum_{i=1}^m \|g^t_{i,\text{probe}}\|^2.$$

Taking conditional expectations and using the selection probabilities for the platform block as in the MSGD lemma,

$$\begin{aligned}\mathbb{E}[\|g^t(\Theta^t)\|^2 \mid \mathscr{F}_t] \le 2\sum_{i=1}^m \mathbb{P}(i_t = i \mid \mathscr{F}_t)\;\mathbb{E}[\|\nabla\ell(z^t,\theta_i^t)\|^2 \mid \mathscr{F}_t, i_t = i] \\ + 2\sum_{j\in U}\mathbb{E}[\|p\,\nabla\ell(\tilde{z}_j^t,\theta_j^t)\|^2 \mid \mathscr{F}_t].\end{aligned}$$

Applying the block bounds gives

$$\begin{aligned}\mathbb{E}[\|g^t(\Theta^t)\|^2 \mid \mathscr{F}_t] &\le 2K_{\text{plat}}\sum_{i=1}^m \mathbb{P}(i_t = i \mid \mathscr{F}_t)\,(1+\|\theta_i^t\|^2) \\ &\quad + 2K_{\text{probe}}\sum_{j\in U}(1+\|\theta_j^t\|^2) \\ &\le 2K_{\text{plat}}\,(1+\|\Theta^t\|^2) \;+\; 2|U|K_{\text{probe}} \;+\; 2K_{\text{probe}}\,\|\Theta^t\|^2 \\ &= 2\big(K_{\text{plat}} + |U|K_{\text{probe}}\big) \;+\; 2\big(K_{\text{plat}} + K_{\text{probe}}\big)\,\|\Theta^t\|^2.\end{aligned}$$

Therefore, setting

$$K := 2\max\{K_{\text{plat}} + |U|K_{\text{probe}},\ K_{\text{plat}} + K_{\text{probe}}\}$$

yields

$$\mathbb{E}[\|v^t\|^2 \mid \mathscr{F}_t] \le K\,(1 + \|\Theta^t\|^2).$$

This completes the proof. □

Lemma B.8 (Gradient of the augmented objective)**.** *For every $i \in [m]$ the gradient of $\widetilde{f}$ with respect to θ_i is*

$$\nabla_{\theta_i} \widetilde{f}(\Theta) = \tau\alpha_i\ \mathbb{E}_{z\sim\mathscr{P}_i}[\nabla\ell(z,\theta_i)] + (1-\tau)\,a_i(\Theta)\ \mathbb{E}_{z\sim\mathscr{D}_i(\Theta)}[\nabla\ell(z,\theta_i)] + p\,\mathbf{1}_{i\in U}\left(\nabla\hat{L}_i(\theta_i) + \lambda\,\theta_i\right).$$

Proof. We treat a single index i; all other coordinates of Θ are held fixed. The term $(1-\tau)a_i(\Theta)\,\mathbb{E}_{z\sim\mathscr{D}_i(\Theta)}[\ell(z,\theta_i)]$ depends on θ_i both explicitly (inside ℓ) and implicitly through $a_i(\Theta)$ and $\mathscr{D}_i(\Theta)$. Lemma 4.3 of Su and Dean [98] proves that for any differentiable ℓ satisfying the stated regularity,

$$\nabla_{\theta_i}\Big(a_i(\Theta)\ \mathbb{E}_{z\sim\mathscr{D}_i(\Theta)}[\ell(z,\theta_i)]\Big) = a_i(\Theta)\ \mathbb{E}_{z\sim\mathscr{D}_i(\Theta)}[\nabla\ell(z,\theta_i)].$$

Multiplying by $(1-\tau)$ yields the corresponding contribution, while the $\tau\alpha_i$-term is immediate. Local L-Lipschitzness of ℓ ensures the required directional limits.

The probing part depends only on θ_i through the finite average $\hat{L}_i(\theta_i) = \frac{1}{n}\sum_{q=1}^{n}\ell(\tilde{z}_i^q,\theta_i)$ and the regularization term $\frac{\lambda}{2}\|\theta_i\|^2$, so $\nabla_{\theta_i}\big(p\,(\hat{L}_i(\theta_i) + \frac{\lambda}{2}\|\theta_i\|^2)\big) = p\,(\nabla\hat{L}_i(\theta_i) + \lambda\,\theta_i)$ if $i \in U$ and zero otherwise. Summing the contributions gives the stated expression. □

Lemma B.9 (Local Lipschitz of $a_i(\Theta)$)**.** *Let the assumptions for Theorem 3.2 hold. Then $a_i(\Theta)$ is locally Lipschitz in Θ.*

Proof. From Lemma B.5, we know that the loss function ℓ is locally Lipschitz. In other words: for every compact set $U \subset \mathbb{R}^{k\times d}$ there is a constant $L_U < \infty$ such that

$$\big|\ell(x,\theta) - \ell(x,\theta')\big| \ \le\ L_U\,\|\theta - \theta'\| \quad \forall\, x \in B(0,R),\ \theta,\theta' \in U.$$

Fix any compact neighborhood U containing both Θ and Θ'. Let

$$p_{\max} \;=\; \sup_{x \in B(0,R)} p(x), \qquad L_U \text{ as above.}$$

We will show $\big|a_i(\Theta) - a_i(\Theta')\big| \le C_U \|\Theta - \Theta'\|$ for some C_U.

Case $m = 2$. With services $i = 1, 2$,

$$a_1(\Theta) - a_1(\Theta') = \int_{X_1(\Theta)\setminus X_1(\Theta')} p(x)\,dx \;-\; \int_{X_1(\Theta')\setminus X_1(\Theta)} p(x)\,dx,$$

so

$$\big|a_1(\Theta) - a_1(\Theta')\big| \;\le\; p_{\max}\Big[\lambda\big(X_1(\Theta) \setminus X_1(\Theta')\big) + \lambda\big(X_1(\Theta') \setminus X_1(\Theta)\big)\Big].$$

For any $x \in X_1(\Theta') \setminus X_1(\Theta)$ we have $\ell(x, \theta_1') < \ell(x, \theta_2')$ and $\ell(x, \theta_2) < \ell(x, \theta_1)$. Hence

$$0 \;<\; \ell(x,\theta_1) - \ell(x,\theta_2) = \big[\ell(x,\theta_1) - \ell(x,\theta_1')\big] + \big[\ell(x,\theta_2') - \ell(x,\theta_2)\big] + \big[\ell(x,\theta_2') - \ell(x,\theta_1')\big],$$

and each difference is bounded by $L_U\|\theta - \theta'\|$. Thus $\ell(x,\theta_1) - \ell(x,\theta_2) \le 2L_U\|\Theta - \Theta'\|$. Define

$$S \;=\; \big\{\, x : \big|\ell(x,\theta_1) - \ell(x,\theta_2)\big| \le 2L_U\|\Theta - \Theta'\| \big\}.$$

Since $\lambda(S) \le (2L_U/C)\|\Theta - \Theta'\|$ for some constant C from Assumption 3.3 and $X_1(\Theta') \setminus X_1(\Theta) \subset S$, we get $\lambda(X_1(\Theta') \setminus X_1(\Theta)) \le C'\|\Theta - \Theta'\|$. The same argument applies to the other set difference, so altogether

$$\big|a_1(\Theta) - a_1(\Theta')\big| \;\le\; 4\,p_{\max}\,L_U\;\|\Theta - \Theta'\|.$$

General m. The same pairwise argument shows for each i and $j \ne i$, $\lambda\big(X_i(\Theta) \cap X_j(\Theta')\big) \le C_U\|\Theta - \Theta'\|$. Summing over all $j \ne i$ gives $\lambda\big(X_i(\Theta) \triangle X_i(\Theta')\big) \le C_U''\|\Theta - \Theta'\|$, and hence $\big|a_i(\Theta) - a_i(\Theta')\big| \le p_{\max}\,C_U''\,\|\Theta - \Theta'\|$.

Since all constants depend only on the compact set U, this proves that $a_i(\Theta)$ is Lipschitz on U, i.e. locally Lipschitz in Θ. □

B.3 Squared Loss: Performance Guarantee

Notation. Let $\{(x_i^q, y_i^q)\}_{q=1}^n$ be the probing sample with true labels y_i^q and pseudo-labels $\tilde{y}_i^q$. Here, the true labels y_i^q are hidden from the learner, and the psuedo-labels $\tilde{y}_i^q$ are observed. For any $\theta \in \mathbb{R}^d$, define

$$\widehat{L}_i(\theta) := \frac{1}{n}\sum_{q=1}^{n} \big(\langle x_i^q, \theta\rangle - \tilde{y}_i^q\big)^2,$$

and

$$\widehat{L}_{i,\mathrm{true}}(\theta) := \frac{1}{n}\sum_{q=1}^{n} \big(\langle x_i^q, \theta\rangle - y_i^q\big)^2.$$

Additionally, define the empirical pseudo–true discrepancy

$$\Delta_i^2 := \frac{1}{n}\sum_{q=1}^{n} \big(\tilde{y}_i^q - y_i^q\big)^2.$$

B.3.1 Proof of Theorem 3.8

Proof of Corollary 3.9. Set $\lambda = \epsilon/\|\theta^\star\|^2$ in Lemma B.10. Define

$$S(\kappa, \lambda) := (4b_\star + 6C_0^2)\sqrt{2\log(2/\kappa)} + 4(Y_{\max} + B_\theta R)B_\theta R + b_{\mathrm{u}}\sqrt{2\log(2/\kappa)},$$

where B_θ and b_{u} are the λ-dependent quantities from Lemma B.10. If

$$n \;\geq\; \underline{n} \;:=\; \frac{S(\kappa,\lambda)^2}{\epsilon^2},$$

then the concentration terms in Lemma B.10 satisfy $S(\kappa,\lambda)/\sqrt{n} \leq \epsilon$. With this choice and $\lambda\|\theta^\star\|^2 = \epsilon$, the explicit bound yields

$$\mathcal{R}(\tilde{\theta}_i) \;\leq\; 6B + \Big(4 + \frac{2}{p}\Big)\epsilon + \epsilon + \epsilon \;=\; 6B + \Big(6 + \frac{2}{p}\Big)\epsilon,$$

which is $O\Big(\big(\frac{p+1}{p}\big)\epsilon + B\Big)$.

To see the stated scaling of $\underline{n}$, note that $B_\theta = \Theta(1/\sqrt{\lambda})$ and $b_{\mathrm{u}} = (Y_{\max} + B_\theta R)^2 = O(1/\lambda)$ with constants depending on $(R, Y_{\max}, M_0, p)$. To surface the dominant dependence on $(R, Y_{\max}, M_0, p)$,

write $A := Y_{\max}^2 + pR^2M_0^2$ so that

$$B_\theta R = R\sqrt{\frac{2A}{\lambda p}} \quad \text{and} \quad (B_\theta R)^2 = \frac{2R^2A}{\lambda p}.$$

The leading terms (in terms of ϵ) in $S(\kappa, \lambda)$ scale as $(B_\theta R)^2$, so one can take

$$\underline{n} = O\left(\frac{R^4\|\theta^\star\|^4}{\epsilon^4}\left(\frac{Y_{\max}^2}{p} + R^2M_0^2\right)^2 \log\frac{1}{\kappa}\right),$$

where we suppress lower-order terms in ϵ coming from $b_\star$ and C_0. □

Theorem 3.8. *Let Assumptions 3.1 - 3.5 hold. For any $\kappa \in (0,1)$, with probability at least $1-\kappa$, every stationary point $\tilde{\Theta}$ of MSGD-P satisfies, for each probing learner i:*

$$\begin{aligned}\mathcal{R}(\tilde{\theta}_i) \;\le\; O\Big(\Big(\frac{p+1}{p}\Big)\epsilon + B + \lambda\|\theta^\star\|^2 \\ + \frac{(p+1)C_{gen}}{p\lambda}\sqrt{\frac{\log(1/\kappa)}{n}}\Big).\end{aligned}$$

where C_{gen} depends on R, $Y_{\max}$, $\|\theta^\star\|$, $\max_{j\neq i}\|\theta_j^0\|$, with explicit form in the Appendix.

Proof. By Lemma B.10, for any $\kappa \in (0,1)$ and with probability at least $1-\kappa$,

$$\mathcal{R}(\tilde{\theta}_i) \;\le\; 6\,B + \Big(4 + \frac{2}{p}\Big)\epsilon + \lambda\,\|\theta^\star\|^2 + \frac{T(\kappa)}{\sqrt{n}},$$

where every $n^{-1/2}$ contribution has been grouped into

$$T(\kappa) := \big(4b_\star + 6C_0^2\big)\sqrt{2\log(2/\kappa)} + b_{\mathrm{u}}\sqrt{2\log(2/\kappa)} + 4(Y_{\max} + B_\theta R)B_\theta R. \tag{B.4}$$

The terms that depend on λ and p are through $B_\theta = \sqrt{2(Y_{\max}^2 + p\,Y_{\max,\mathrm{probe}}^2)/(\lambda\,p)}$, which also appears inside $b_{\mathrm{u}} = (Y_{\max} + B_\theta R)^2$. Expanding the B_θ-dependent pieces, from (B.4),

$$T(\kappa) = \big(4b_\star + 6C_0^2\big)\sqrt{2\log(2/\kappa)} + \underbrace{\Big[(Y_{\max} + B_\theta R)^2\sqrt{2\log(2/\kappa)} + 4(Y_{\max} + B_\theta R)B_\theta R\Big]}_{B_\theta\text{-dependent}}.$$

Thus the λ-independent part is already $O\big(\sqrt{\log(1/\kappa)}\big)$, so it remains to control the bracketed term.

Expanding gives

$$(Y_{\max} + B_\theta R)^2 \sqrt{2\log(2/\kappa)} + 4(Y_{\max} + B_\theta R) B_\theta R = O\Big(\sqrt{\log(1/\kappa)}\,\big[Y_{\max}^2 + Y_{\max} B_\theta R + B_\theta^2 R^2\big]\Big).$$

Plugging in the value of B_θ from Lemma B.15 and assuming $\lambda < 1$, we can combine with the (λ, p)-independent part of $T(\kappa)$ to get

$$C_{\text{gen}} = R(Y_{\max}^2 + Y_{\max,\text{probe}}^2) + 4b_\star + 6C_0^2$$

□

Lemma B.10 (Squared-loss performance Full Statement)**.** *Let Assumption 3.5 hold with parameter B. Then, for any $\kappa \in (0,1)$, with probability at least $1-\kappa$ over the probing sample, every stationary point $\tilde{\Theta}$ of MSGD-P satisfies, for probing learner $i \in [m]$,*

$$\begin{aligned}\mathcal{R}(\tilde{\theta}_i) \;\le\; & 6\,B + \Big(4 + \frac{2}{p}\Big)\,\epsilon + \lambda\,\|\theta^\star\|^2 \\ & + \frac{(4\,b_\star + 6\,C_0^2)\,\sqrt{2\log(2/\kappa)}}{\sqrt{n}} + \frac{4\,(Y_{\max} + B_\theta R)\,B_\theta R}{\sqrt{n}} + b_{\text{u}}\sqrt{\frac{2\log(2/\kappa)}{n}},\end{aligned}$$

where the constants are defined as follows:

- $C_0 \;:=\; Y_{\max} + Y_{\max,\text{probe}}$.
- $B_\theta \;:=\; \sqrt{\dfrac{2(Y_{\max}^2 + p\,Y_{\max,\text{probe}}^2)}{\lambda\,p}}$ *is a radius (independent of $\theta^\star$) with* $Y_{\max,\text{probe}} \;:=\; R\,M_0$, $M_0 \;:=\; \max\limits_{j \neq i} \|\theta_j^0\|$, *which bounds the pseudo-label magnitudes via* $|\tilde{y}_i^q| \le Y_{\max,\text{probe}}$.
- $b_\star \;:=\; \big(Y_{\max} + R\,\|\theta^\star\|\big)^2$.
- $b_{\text{u}} \;:=\; \big(Y_{\max} + B_\theta R\big)^2$.

Proof. We work on the event $\mathcal{E}_{\text{u}} \cap \mathcal{E}_\star$ where $\mathcal{E}_{\text{u}}$ is the event of Lemma B.13 (probability $\ge 1 - \kappa/2$ with B_θ from Lemma B.15) and $\mathcal{E}_\star$ is the event of Lemma B.12 (probability $\ge 1 - \kappa/2$). By a union bound,

$$\mathbb{P}(\mathcal{E}_{\text{u}} \cap \mathcal{E}_\star) \ge 1 - \kappa.$$

Bound on $\widehat{L}_i(\tilde{\theta}_i)$. By stationarity, $\tilde{\theta}_i$ minimizes

$$\hat{\Phi}_i^{\text{probe}}(\theta;\tilde{\Theta}) = \tau\alpha_i\ \mathbb{E}_{\mathscr{P}_i}[\ell(z,\theta)] + (1-\tau)a_i(\tilde{\Theta})\ \mathbb{E}_{\mathscr{D}_i(\tilde{\Theta})}[\ell(z,\theta)] + p\,\widehat{L}_i(\theta) + \tfrac{\lambda p}{2}\|\theta\|^2.$$

Thus, comparing the objective at $\tilde{\theta}_i$ and $\theta^\star$,

$$p\,\widehat{L}_i(\tilde{\theta}_i)\ \le\ \tau\alpha_i\ \mathbb{E}_{\mathscr{P}_i}[\ell(z,\theta^\star)] + (1-\tau)a_i(\tilde{\Theta})\ \mathbb{E}_{\mathscr{D}_i(\tilde{\Theta})}[\ell(z,\theta^\star)] + p\,\widehat{L}_i(\theta^\star) + \tfrac{\lambda p}{2}\|\theta^\star\|^2.$$

Since

$$\tau\alpha_i\,\mathbb{E}_{\mathscr{P}_i}\,\ell(\theta^\star) + (1-\tau)a_i(\tilde{\Theta})\,\mathbb{E}_{\mathscr{D}_i(\tilde{\Theta})}\,\ell(\theta^\star) \le \epsilon,$$

dividing by p, we obtain

$$\widehat{L}_i(\tilde{\theta}_i)\ \le\ \frac{\epsilon}{p}\ +\ \widehat{L}_i(\theta^\star)\ +\ \frac{\lambda}{2}\,\|\theta^\star\|^2.$$

By Lemma B.11 and Lemma B.12,

$$\widehat{L}_i(\theta^\star) \le 2\,\widehat{L}_{i,\text{true}}(\theta^\star) + 2\,\Delta_i^2 \le 2\,\epsilon + 2\,\Lambda_\star + 2\,\Delta_i^2.$$

Hence

$$\widehat{L}_i(\tilde{\theta}_i)\ \le\ \frac{\epsilon}{p} + 2\,\epsilon + 2\,\Lambda_\star + 2\,\Delta_i^2\ +\ \frac{\lambda}{2}\,\|\theta^\star\|^2.$$

Bound on $\widehat{L}_{i,\text{true}}(\tilde{\theta}_i)$. Applying Lemma B.11 again gives

$$\widehat{L}_{i,\text{true}}(\tilde{\theta}_i)\ \le\ 2\,\widehat{L}_i(\tilde{\theta}_i) + 2\,\Delta_i^2.$$

Substituting the bound from Step 1,

$$\widehat{L}_{i,\text{true}}(\tilde{\theta}_i)\ \le\ \Big(\tfrac{2}{p} + 4\Big)\epsilon\ +\ 4\,\Lambda_\star\ +\ 6\,\Delta_i^2\ +\ \lambda\,\|\theta^\star\|^2.$$

By Lemma B.14, on $\mathcal{E}_\star$,

$$\Delta_i^2 \le B + C_0^2\,\sqrt{\frac{2\log(2/\kappa)}{n}}.$$

Therefore,

$$\widehat{L}_{i,\mathrm{true}}(\tilde{\theta}_i) \;\le\; 6\,B \;+\; \Big(\tfrac{2}{p}+4\Big)\epsilon \;+\; 4\,\Lambda_\star \;+\; 6\,C_0^2\sqrt{\frac{2\log(2/\kappa)}{n}} \;+\; \lambda\,\|\theta^\star\|^2.$$

Bound on $\mathcal{R}(\tilde{\theta}_i)$. Finally, on $\mathcal{E}_{\mathrm{u}}$ (with $\|\tilde{\theta}_i\| \le B_\theta$ from Lemma B.15),

$$\mathcal{R}(\tilde{\theta}_i) \le \widehat{L}_{i,\mathrm{true}}(\tilde{\theta}_i) + \Lambda_{\mathrm{u}}.$$

□

Alternative scaling with explicit λ. If we keep λ explicit and only choose n to control the concentration terms in Lemma B.10, then for any target tolerance $\delta > 0$ it suffices to take

$$n \;\ge\; \bar{n}(\lambda,\delta) \;:=\; \frac{S(\kappa,\lambda)^2}{\delta^2},$$

which yields

$$\mathcal{R}(\tilde{\theta}_i) \;\le\; 6B + \Big(4+\frac{2}{p}\Big)\epsilon + \lambda\|\theta^\star\|^2 + \delta.$$

Writing $Y_{\max,\mathrm{probe}} = RM_0$ and $A := Y_{\max}^2 + pR^2M_0^2$, we have

$$B_\theta = \sqrt{\frac{2A}{\lambda p}} \quad \text{and} \quad (Y_{\max} + B_\theta R)^2 \le 2Y_{\max}^2 + \frac{4R^2A}{\lambda p}.$$

Using $b_\star = (Y_{\max} + R\|\theta^\star\|)^2$ and $C_0 = Y_{\max} + RM_0$, this implies

$$S(\kappa,\lambda)^2 = O\!\left(\log\frac{1}{\kappa}\right)\left[(Y_{\max}+R\|\theta^\star\|)^4 + (Y_{\max}+RM_0)^4 + \frac{R^4A^2}{\lambda^2p^2}\right],$$

and therefore a sufficient choice is

$$n = O\!\left(\frac{\log(1/\kappa)}{\delta^2}\left[(Y_{\max}+R\|\theta^\star\|)^4 + (Y_{\max}+RM_0)^4 + \frac{R^4(Y_{\max}^2+pR^2M_0^2)^2}{\lambda^2p^2}\right]\right).$$

Setting $\delta = \epsilon$ yields the same bias expression as above, but now with n explicit in $(R, Y_{\max}, M_0, p, \lambda, \epsilon, \kappa)$.

□

Lemma B.11. *We have*

$$\widehat{L}_{i,\mathrm{true}}(\theta) \;\le\; \Big(\sqrt{\widehat{L}_i(\theta)} + \Delta_i\Big)^2 \quad \textit{and} \quad \widehat{L}_i(\theta) \;\le\; \Big(\sqrt{\widehat{L}_{i,\mathrm{true}}(\theta)} + \Delta_i\Big)^2.$$

In particular, $\widehat{L}_{i,\mathrm{true}}(\theta) \le 2\,\widehat{L}_i(\theta) + 2\,\Delta_i^2$ *and* $\widehat{L}_i(\theta) \le 2\,\widehat{L}_{i,\mathrm{true}}(\theta) + 2\,\Delta_i^2$.

Proof of Lemma B.11. Let $X \in \mathbb{R}^{n\times d}$ be the design matrix, $\mathbf{a} := X\theta - \tilde{\mathbf{y}}$ and $\mathbf{b} := \tilde{\mathbf{y}} - \mathbf{y}$. Then

$$X\theta - \mathbf{y} = \mathbf{a} + \mathbf{b}.$$

By the triangle inequality,

$$\sqrt{\widehat{L}_{i,\mathrm{true}}(\theta)} = \frac{\|X\theta - \mathbf{y}\|}{\sqrt{n}} \le \frac{\|\mathbf{a}\|}{\sqrt{n}} + \frac{\|\mathbf{b}\|}{\sqrt{n}} = \sqrt{\widehat{L}_i(\theta)} + \Delta_i.$$

Squaring yields the first inequality; the second is analogous. □

Lemma B.12 (One–point deviation at $\theta^\star$)**.** *For any* $\kappa \in (0,1)$*, with probability at least* $1 - \kappa/2$*,*

$$\big|\widehat{L}_{i,\mathrm{true}}(\theta^\star) - \mathcal{R}(\theta^\star)\big| \;\le\; \Lambda_\star \quad \textit{where} \quad \Lambda_\star \;:=\; b_\star \sqrt{\frac{2\log(2/\kappa)}{n}}, \quad b_\star \;:=\; \big(Y_{\max} + R\,\|\theta^\star\|\big)^2.$$

Proof of Lemma B.12. For each q, define the random variable

$$Z_q := (\langle \tilde{x}_i^q, \theta^\star\rangle - y_i^q)^2.$$

Since $|y| \le Y_{\max}$ and $\|x\| \le R$ a.s., we have

$$Z_q \in [0,\ (Y_{\max} + R\|\theta^\star\|)^2] = [0, b_\star].$$

By Hoeffding's inequality, with probability at least $1 - \kappa/2$,

$$\left|\frac{1}{n}\sum_{q=1}^{n} Z_q - \mathbb{E}[Z]\right| \le b_\star \sqrt{\frac{2\log(2/\kappa)}{n}}.$$

□

Lemma B.13 (Uniform deviation over $\{\|\theta\| \le B_\theta\}$)**.** *Fix any radius $B_\theta > 0$ and let $\kappa \in (0,1)$. Define the squared-loss class*

$$\mathcal{F}_{B_\theta} \;:=\; \Big\{\, f_\theta(x,y) \;=\; \big(y - \langle x,\theta\rangle\big)^2 \;\Big|\; \|\theta\| \le B_\theta \Big\}.$$

Then, with probability at least $1 - \kappa/2$ (over the draw of the probing sample of size n),

$$\sup_{\|\theta\|\le B_\theta} \big|\widehat{L}_{i,\mathrm{true}}(\theta) - \mathcal{R}(\theta)\big| \;\le\; \Lambda_{\mathrm{u}},$$

where

$$\Lambda_{\mathrm{u}} \;:=\; \frac{4\,(Y_{\max} + B_\theta R)\,B_\theta R}{\sqrt{n}} \;+\; b_{\mathrm{u}}\sqrt{\frac{2\log(2/\kappa)}{n}}, \qquad b_{\mathrm{u}} \;:=\; (Y_{\max} + B_\theta R)^2.$$

Proof. We apply Wainwright's Theorem 4.10.

Uniform bound b for $\mathcal{F}_{B_\theta}$. For any $\|\theta\| \le B_\theta$ and any (x,y) with $\|x\| \le R$, $|y| \le Y_{\max}$, we have

$$\big|y - \langle x,\theta\rangle\big| \;\le\; |y| + |\langle x,\theta\rangle| \;\le\; Y_{\max} + \|x\|\,\|\theta\| \;\le\; Y_{\max} + B_\theta R.$$

Hence

$$0 \;\le\; f_\theta(x,y) \;=\; \big(y - \langle x,\theta\rangle\big)^2 \;\le\; (Y_{\max} + B_\theta R)^2 \;=:\; b_{\mathrm{u}}.$$

Thus $\mathcal{F}_{B_\theta}$ is b_{u}–uniformly bounded.

Bound $R_n(\mathcal{F}_{B_\theta})$ via contraction to the linear class. Fix a sample $S = \{(x_i,y_i)\}_{i=1}^n$ and let $\sigma_1,\dots,\sigma_n$ be i.i.d. Rademacher signs. The empirical Rademacher average of $\mathcal{F}_{B_\theta}$ on S is

$$\mathrm{Rad}_S(\mathcal{F}_{B_\theta}) \;:=\; \mathbb{E}_\sigma\Big[\sup_{\|\theta\|\le B_\theta} \frac{1}{n}\sum_{i=1}^n \sigma_i\big(y_i - \langle x_i,\theta\rangle\big)^2\Big].$$

Define $u_{i,\theta} := \langle x_i,\theta\rangle$ and, for each i, the recentered function

$$\psi_i(u) \;:=\; \big(y_i - u\big)^2 - \big(y_i - 0\big)^2 \;=\; u^2 - 2\,y_i\,u, \qquad \psi_i(0) = 0.$$

Since $\mathbb{E}[\sigma_i] = 0$, the constant offsets $\{(y_i - 0)^2\}$ vanish in the Rademacher average. Thus

$$\mathrm{Rad}_S(\mathcal{F}_{B_\theta}) \;=\; \mathbb{E}_\sigma \Big[\sup_{\|\theta\| \le B_\theta} \; \frac{1}{n} \sum_{i=1}^{n} \sigma_i \, \psi_i\big(u_{i,\theta}\big) \Big].$$

We now bound the Lipschitz constants of ψ_i over the relevant range. For any $u, v \in [-B_\theta R, B_\theta R]$,

$$\big|\psi_i(u) - \psi_i(v)\big| = \big|\,(u-v)\,\big[(u-y_i)+(v-y_i)\big]\,\big| \;\le\; |u-v|\,\big(\,|u-y_i|+|v-y_i|\,\big) \;\le\; 2(Y_{\max}+B_\theta R)\,|u-v|.$$

Hence each ψ_i is L–Lipschitz with

$$L \;:=\; 2(Y_{\max} + B_\theta R), \qquad \psi_i(0) = 0.$$

By the Ledoux–Talagrand contraction inequality (applied elementwise to $\{\psi_i\}$ with common Lipschitz constant L and $\psi_i(0) = 0$), we have

$$\mathrm{Rad}_S(\mathcal{F}_{B_\theta}) \;\le\; L \cdot \mathbb{E}_\sigma \Big[\sup_{\|\theta\| \le B_\theta} \; \frac{1}{n} \sum_{i=1}^{n} \sigma_i \, u_{i,\theta} \Big] \;=\; L \cdot \mathrm{Rad}_S(\mathcal{G}_{B_\theta}),$$

where $\mathcal{G}_{B_\theta} := \{(x, y) \mapsto \langle x, \theta \rangle : \; \|\theta\| \le B_\theta\}$ is the linear class (note: dependence on y disappears).

The empirical Rademacher average of $\mathcal{G}_{B_\theta}$ on S is standard:

$$\mathrm{Rad}_S(\mathcal{G}_{B_\theta}) = \mathbb{E}_\sigma \Big[\sup_{\|\theta\| \le B_\theta} \; \frac{1}{n} \sum_{i=1}^{n} \sigma_i \, \langle x_i, \theta \rangle \Big] = \frac{B_\theta}{n} \; \mathbb{E}_\sigma \Big\| \sum_{i=1}^{n} \sigma_i x_i \Big\| \;\le\; \frac{B_\theta}{n} \sqrt{\mathbb{E}_\sigma \Big\| \sum_{i=1}^{n} \sigma_i x_i \Big\|^2} = \frac{B_\theta}{n} \sqrt{\sum_{i=1}^{n} \|x_i\|^2}.$$

Using $\|x_i\| \le R$, we conclude $\mathrm{Rad}_S(\mathcal{G}_{B_\theta}) \le B_\theta R / \sqrt{n}$. Combining,

$$\mathrm{Rad}_S(\mathcal{F}_{B_\theta}) \;\le\; L \cdot \mathrm{Rad}_S(\mathcal{G}_{B_\theta}) \;\le\; 2(Y_{\max} + B_\theta R) \, \frac{B_\theta R}{\sqrt{n}}.$$

Taking expectation over the sample (or simply noting that this bound holds for any S satisfying $\|x_i\| \le R$) yields

$$R_n(\mathcal{F}_{B_\theta}) \;\le\; 2(Y_{\max} + B_\theta R) \, \frac{B_\theta R}{\sqrt{n}}.$$

Apply Theorem 4.10 and choose δ. By Theorem 4.10, for any $\delta > 0$,

$$\sup_{\|\theta\| \le B_\theta} \left| \widehat{L}_{i,\text{true}}(\theta) - \mathcal{R}(\theta) \right| \;\le\; 2\, R_n(\mathcal{F}_{B_\theta}) + \delta \;\le\; \frac{4(Y_{\max} + B_\theta R)\, B_\theta R}{\sqrt{n}} + \delta,$$

with probability at least $1 - \exp\big(-\frac{n\,\delta^2}{2b_{\text{u}}^2}\big)$. Set $\delta = b_{\text{u}} \sqrt{\frac{2\log(2/\kappa)}{n}}$ so that $\exp\big(-\frac{n\,\delta^2}{2b_{\text{u}}^2}\big) = \exp\big(-\log(2/\kappa)\big) = \kappa/2$. The stated deviation bound Λ_{u} follows. □

Lemma B.14 (Concentration of Δ_n^2 under Accurate Probing)**.** *Assume that $|\tilde{y}| \le Y_{\max,\text{probe}}$ almost surely (e.g., $Y_{\max,\text{probe}} = R \max_j \|\theta_j^0\|$ for linear peers at initialization). Then for any $\kappa \in (0,1)$, with probability at least $1 - \kappa/2$,*

$$\Delta_n^2 \;\le\; B \;+\; C_0^2 \sqrt{\frac{2\log(2/\kappa)}{n}}, \qquad \textit{where} \quad C_0 \;:=\; Y_{\max} + Y_{\max,\text{probe}}.$$

Proof. Let $W = (\tilde{y} - y)^2$. By the boundedness assumptions, $0 \le W \le (Y_{\max} + Y_{\max,\text{probe}})^2 =: U$ almost surely, and $\mathbb{E}[W] \le B$ by Accurate Probing. By Hoeffding's inequality,

$$\Pr\left(\frac{1}{n} \sum_{q=1}^{n} W_q - \mathbb{E}[W] \ge t \right) \;\le\; \exp\!\left(-\frac{2nt^2}{U^2} \right).$$

Choosing $t = U\sqrt{\frac{2\log(2/\kappa)}{n}}$ yields $\Pr\big(\Delta_n^2 \ge \mathbb{E}[W] + U\sqrt{\frac{2\log(2/\kappa)}{n}}\big) \le \kappa/2$. Using $\mathbb{E}[W] \le B$ gives the stated bound with $C_0^2 = U$. □

Lemma B.15 (A priori norm bound and choice of B_θ)**.** *For any probing learner $i \in U$,*

$$\|\tilde{\theta}_i\| \;\le\; B_\theta \quad \textit{with} \quad B_\theta \;:=\; \sqrt{\frac{2(Y_{\max}^2 + p\, Y_{\max,\text{probe}}^2)}{\lambda\, p}}.$$

Proof of Lemma B.15. At any stationary point $\tilde{\Theta}$, $\tilde{\theta}_i$ minimizes the convex function

$$\hat{\Phi}_i^{\text{probe}}(\theta; \tilde{\Theta}) \;:=\; \tau\alpha_i\, \mathbb{E}_{z\sim\mathcal{P}_i}\big[\ell(z,\theta)\big] \;+\; (1-\tau)\, a_i(\tilde{\Theta})\, \mathbb{E}_{z\sim\mathcal{D}_i(\tilde{\Theta})}\big[\ell(z,\theta)\big] \;+\; p\, \widehat{L}_i(\theta) \;+\; \frac{\lambda\, p}{2}\, \|\theta\|^2.$$

By optimality of $\tilde{\theta}_i$ for $\hat{\Phi}_i^{\text{probe}}(\cdot; \tilde{\Theta})$,

$$\hat{\Phi}_i^{\text{probe}}(\tilde{\theta}_i; \tilde{\Theta}) \le \hat{\Phi}_i^{\text{probe}}(0; \tilde{\Theta}).$$

Using squared loss and $|y| \le Y_{\max}$, we have

$$\widehat{L}_i(0) = \frac{1}{n}\sum_{q=1}^{n}(\tilde{y}_i^q)^2 \le Y_{\max,\text{probe}}^2.$$

Hence,

$$\frac{\lambda p}{2}\,\|\tilde{\theta}_i\|^2 \;\le\; \tau\alpha_i\, Y_{\max}^2 \;+\; (1-\tau)\, a_i(\tilde{\Theta})\, Y_{\max}^2 \;+\; p\,\widehat{L}_i(0) \;\le\; Y_{\max}^2 + p\, Y_{\max,\text{probe}}^2.$$

Thus

$$\|\tilde{\theta}_i\| \;\le\; \sqrt{\frac{2(Y_{\max}^2 + p\, Y_{\max,\text{probe}}^2)}{\lambda\, p}} \;=\; B_\theta.$$

□

B.4 Cross-Entropy Loss: Performance Guarantee

Notation. In this section, we provide a high-probability upper bound on the full-population cross-entropy risk for a probing learner in MSGD-P at a stationary point. The proof mirrors the squared-loss analysis, with key differences arising from the geometry of the softmax and cross-entropy.

Throughout this section, let K denote the number of classes. Let $\theta \equiv W \in \mathbb{R}^{K\times d}$ be the matrix of class-wise linear weights and define logits $z(x) = Wx \in \mathbb{R}^K$ and predicted probabilities $q_W(x) = \text{softmax}(z(x))$. The multiclass cross-entropy loss is

$$\ell_{\text{CE}}((x,y),W) \;=\; -\sum_{c=1}^{K} y_c \,\log q_W(x)_c \;=\; -\log q_W(x)_Y,$$

where $y \in \{e_1,\dots,e_K\}$ is one-hot and Y is the true class index. We assume $\|x\| \le R$ almost surely.

For a probing learner $i \in U$ (with probing weight $p > 0$), we define its augmented objective at Θ (as per MSGD-P) by

$$\begin{aligned}\Phi_i(W;\Theta) \;=\;& \tau\alpha_i\, \mathbb{E}_{(x,y)\sim\mathcal{P}_i}\left[\ell_{\text{CE}}((x,y),W)\right] \\ &+\; (1-\tau)\, a_i(\Theta)\, \mathbb{E}_{(x,y)\sim\mathcal{D}_i(\Theta)}\left[\ell_{\text{CE}}((x,y),W)\right] \\ &+\; p\,\widehat{L}_{\text{probe}}(W) \;+\; \frac{\lambda p}{2}\,\|W\|_F^2,\end{aligned}$$

where

$$\widehat{L}_{\text{probe}}(W) \;=\; \frac{1}{n}\sum_{q=1}^{n}\ell_{\text{CE}}\big((\tilde{x}_i^q,\tilde{y}_i^q),W\big) \;=\; -\frac{1}{n}\sum_{q=1}^{n}\sum_{c=1}^{K}\tilde{y}_{i,c}^q\log q_W(\tilde{x}_i^q)_c.$$

Here $\{(\tilde{x}_i^q)\}_{q=1}^n$ are probing covariates drawn i.i.d. from $\mathcal{P}_X$, and $\{\tilde{y}_i^q\}_{q=1}^n$ are soft pseudo-labels. For analysis, let y_i^q be the (hidden) true labels of $\tilde{x}_i^q$, and define the empirical "true" probing CE:

$$\widehat{L}_{\text{true}}(W) \;=\; \frac{1}{n}\sum_{q=1}^{n}\ell_{\text{CE}}\big((\tilde{x}_i^q,y_i^q),W\big) \;=\; -\frac{1}{n}\sum_{q=1}^{n}\log q_W(\tilde{x}_i^q)_{y_i^q}.$$

Let $\theta^\star \in \arg\min_W \mathcal{R}(W)$ be a (finite-norm) population minimizer of the unregularized CE risk $\mathcal{R}(W)=\mathbb{E}[\ell_{\text{CE}}((x,y),W)]$, with $\epsilon := \mathcal{R}(\theta^\star)$ and $\|\theta^\star\|_F =: M_\star < \infty$.

Aggregation Rule. we define peer logits $z_j(x,\Theta)=\theta_j^0 x \in \mathbb{R}^K$ and the coordinatewise median of logits

$$\tilde{z}_c(x,\Theta) := \text{median}(z_{j,c}(x,\Theta) : j \in T_i(x)), \quad c \in [K].$$

Then, we set the probing label as $y_{\text{agg},i}(x,\Theta)=\text{softmax}(\tilde{z}(x,\Theta))$.

B.4.1 Accurate Probing Assumption and Proofs

Scenario	$T_i(x)$	B
Majority-good	$[m]$	$\epsilon + 2Rr$
Market-leader	$\{j^*\}$	ξ
Partial knowledge	G	$\epsilon + 2Rr$
Preference-aware	$\{\pi(x)\}$	ϵ

Table B.1: Probing accuracy parameters for cross-entropy loss.

Assumption B.1 (Accurate Probing for CE)**.** *There exists $B_{\text{CE}} \geq 0$ such that*

$$\mathbb{E}\left[\text{CE}(y,\tilde{y})\right] \;=\; \mathbb{E}\left[-\log\tilde{y}_Y\right] \;\leq\; B_{\text{CE}},$$

where the expectation is over $(x,y)\sim\mathcal{P}$ and $\tilde{y}=\tilde{y}(x)$ produced by the robust aggregator.

We now present sufficient conditions ensuring Assumption B.1.

Lemma B.16 (Sufficient conditions for Accurate Probing (CE))**.** *Assumption B.1 holds in each of the following scenarios:* ***(i) Majority-good.*** *Suppose strictly more than half of peers satisfy*

$\|\theta_j^0 - \theta^\star\|_F \le r$. *Then*

$$B_{\mathrm{CE}} \;=\; \mathbb{E}\big[-\log \tilde{y}_Y\big] \;\le\; \epsilon \;+\; 2\,R\,r.$$

(ii) Market-leader. *Suppose learner i probes a single peer $j^\star$ with $\mathbb{E}[\ell_{\mathrm{CE}}((x,y),\theta_{j^\star}^0)] \le \xi$. If $\tilde{y}(x) = q_{\theta_{j^\star}^0}(x)$, then $B_{\mathrm{CE}} \le \xi$.*

(iii) Partial knowledge. *If a known subset G of peers satisfies $|G| > (m-1)/2$ and $\|\theta_j^0 - \theta^\star\|_F \le r$ for all $j \in G$, and the aggregator uses the coordinatewise median over G, the same bound as in case (i) holds.*

(iv) Preference-aware. *Suppose each peer j solves the ERM on its preference partition $\mathscr{P}_j$:*

$$\bar{\theta}_j \in \arg\min_W \; \mathbb{E}_{(x,y)\sim\mathscr{P}_j}\big[\ell_{\mathrm{CE}}((x,y),W)\big].$$

If learner i probes $\tilde{y}(x) = q_{\bar{\theta}_{\pi(x)}}(x)$, then

$$B_{\mathrm{CE}} \;\le\; \sum_{j=1}^{K} \alpha_j \; \mathbb{E}_{\mathscr{P}_j}\big[\ell_{\mathrm{CE}}((x,y),\bar{\theta}_j)\big] \;\le\; \epsilon.$$

Proof. We treat each scenario in turn.

(i) Majority-good. Fix any x with $\|x\| \le R$ and write $z^\star(x) = \theta^\star x$. For any good peer j (i.e., $\|\theta_j^0 - \theta^\star\|_F \le r$) and any class c,

$$|z_{j,c}(x) - z_c^\star(x)| \;=\; |(\theta_{j,c}^0 - \theta_c^\star)^\top x| \;\le\; \|\theta_{j,c}^0 - \theta_c^\star\|_2 \cdot \|x\| \;\le\; \|\theta_j^0 - \theta^\star\|_F \cdot \|x\| \;\le\; r\,R.$$

Since strictly more than half of the $\{z_{j,c}(x)\}_{j\neq i}$ lie in $[z_c^\star(x) - rR,\; z_c^\star(x) + rR]$, their coordinatewise median must also lie in this interval. Hence

$$\|\tilde{z}(x) - z^\star(x)\|_\infty \;\le\; r\,R.$$

Let $f(z) = -\log \mathrm{softmax}(z)_Y$. By Lemma B.17,

$$-\log \tilde{y}_Y \;=\; f(\tilde{z}(x)) \;\le\; f\big(z^\star(x)\big) + 2\,\|\tilde{z}(x) - z^\star(x)\|_\infty \;\le\; -\log q_Y^\star(x) + 2\,r\,R,$$

where $q^\star = \mathrm{softmax}(z^\star)$. Taking expectations over $(x, y) \sim \mathscr{P}$, we get

$$B_{\mathrm{CE}} \;=\; \mathbb{E}\big[-\log \tilde{y}_Y\big] \;\le\; \mathbb{E}\big[-\log q^\star_Y(x)\big] + 2\,r\,R \;=\; \epsilon + 2\,r\,R.$$

(ii) Market-leader. Direct: $B_{\mathrm{CE}} = \mathbb{E}[-\log \tilde{y}_Y] = \mathbb{E}[\ell_{\mathrm{CE}}((x, y), \theta^0_{j^\star})] \le \xi$.

(iii) Partial knowledge. The proof is identical to case (i), restricting the median to G. Since $|G| > (m-1)/2$ and all learners in G satisfy $\|\theta^0_j - \theta^\star\|_F \le r$, the coordinatewise median over G satisfies the same bound.

(iv) Preference-aware. By optimality of $\bar{\theta}_j$ for the ERM objective,

$$\mathbb{E}_{\mathscr{P}_j}\big[\ell_{\mathrm{CE}}((x, y), \bar{\theta}_j)\big] \;\le\; \mathbb{E}_{\mathscr{P}_j}\big[\ell_{\mathrm{CE}}((x, y), \theta^\star)\big].$$

Summing over j with weights α_j gives

$$B_{\mathrm{CE}} \;\le\; \sum_j \alpha_j\, \mathbb{E}_{\mathscr{P}_j}\, \ell_{\mathrm{CE}}(\bar{\theta}_j) \;\le\; \sum_j \alpha_j\, \mathbb{E}_{\mathscr{P}_j}\, \ell_{\mathrm{CE}}(\theta^\star) \;=\; \epsilon.$$

□

Lemma B.17 (CE is 2-Lipschitz in logits under $\|\cdot\|_\infty$)**.** *Fix a class $Y \in [K]$ and define $f(z) := -\log \mathrm{softmax}(z)_Y = -z_Y + \mathrm{logsumexp}(z)$ for $z \in \mathbb{R}^K$. Then for any $z, z' \in \mathbb{R}^K$,*

$$\big|f(z') - f(z)\big| \;\le\; 2\,\|z' - z\|_\infty.$$

Proof. We have $\nabla f(z) = q(z) - e_Y$, where $q(z) = \mathrm{softmax}(z)$ and e_Y is the one-hot at Y. By the mean value inequality with dual norms $(\|\cdot\|_\infty, \|\cdot\|_1)$,

$$|f(z') - f(z)| \;\le\; \sup_{\xi \in [z, z']} \|\nabla f(\xi)\|_1 \cdot \|z' - z\|_\infty \;=\; \sup_\xi \sum_{k=1}^K |q_k(\xi) - (e_Y)_k| \cdot \|z' - z\|_\infty.$$

Since $y = e_Y$ is one-hot and $q \in \Delta^{K-1}$, $\sum_{k=1}^K |q_k - (e_Y)_k| = |q_Y - 1| + \sum_{k \ne Y} q_k = (1 - q_Y) + (1 - q_Y) \le 2$. Hence $|f(z') - f(z)| \le 2\,\|z' - z\|_\infty$. □

B.4.2 Performance Bound

We will use two probability floors:

$$\gamma_\star \;:=\; \frac{1}{K}\,\exp\!\big(-2R\,\|\theta^\star\|_F\big), \quad \Gamma_\star \;:=\; \log\frac{1}{\gamma_\star} \;=\; \log K + 2R\|\theta^\star\|_F,$$

and

$$B_\theta \;:=\; \sqrt{\frac{2(1+p)\log K}{\lambda\, p}}, \qquad \gamma_B \;:=\; \frac{1}{K}\,\exp\!\big(-2R\,B_\theta\big), \qquad \Gamma_B \;:=\; \log\frac{1}{\gamma_B} \;=\; \log K + 2RB_\theta.$$

Corollary B.18 (Big-O summary)**.** *Under the assumptions of Theorem B.19, with probability at least* $1-\kappa$*,*

$$\mathcal{R}(\tilde\theta_i) \le O\Big(\Big(\frac{p+1}{p}\Big)\epsilon \;+\; \lambda\,\|\theta^\star\|_F^2 \;+\; C_{bias}\,\sqrt{B_{\mathrm{CE}}} \;+\; \frac{(p+1)\,C_{gen}}{p}\,\sqrt{\tfrac{\log(1/\kappa)}{\lambda n}}\Big),$$

where the constant $C_{bias} = R\left(\|\theta^\star\|_F + \sqrt{2\log K/\lambda}\right)$ *and* $C_{gen} = C_{gen}(R, \|\theta^\star\|, \log(K))$ *is specified in the appendix.*

Theorem B.19 (CE performance bound with probing)**.** *Let Assumptions 3.1, 3.2, 3.3, 3.4, and B.1 hold. Let* $\tilde\Theta$ *be a stationary point of MSGD-P and let* $\tilde\theta_i$ *be the parameter for a probing learner* $i \in U$*. Then for any* $\kappa \in (0,1)$*, with probability at least* $1-\kappa$ *over the probing sample,*

$$\begin{aligned}\mathcal{R}(\tilde\theta_i) \le \Big(1+\frac{1}{p}\Big)\epsilon + \frac{\lambda}{2}\|\theta^\star\|_F^2 + R(\|\theta^\star\|_F + B_\theta)\sqrt{2B_{\mathrm{CE}}} + 4R\sqrt{\frac{(1+p)K\log K}{\lambda\, p\, n}} \\ + \big(\Gamma_\star + \Gamma_B + R(\|\theta^\star\|_F + B_\theta)\big)\sqrt{\frac{\log(3/\kappa)}{2n}}.\end{aligned}$$

Proof. We work on the event $\mathcal{E}_{\mathrm{pinsker}} \cap \mathcal{E}_\star \cap \mathcal{E}_{\mathrm{u}}$ where:

- $\mathcal{E}_{\mathrm{pinsker}}$ is the event of Lemma B.22 (probability $\ge 1-\kappa/3$),
- $\mathcal{E}_\star$ is the event of Lemma B.23 (probability $\ge 1-\kappa/3$),
- $\mathcal{E}_{\mathrm{u}}$ is the event of Lemma B.24 (probability $\ge 1-\kappa/3$).

By a union bound,

$$\mathbb{P}(\mathcal{E}_{\mathrm{pinsker}} \cap \mathcal{E}_\star \cap \mathcal{E}_{\mathrm{u}}) \ge 1-\kappa.$$

Bound on $\widehat{L}_{\text{probe}}(\tilde{\theta}_i)$. By Lemma B.20, dividing by p,

$$\widehat{L}_{\text{probe}}(\tilde{\theta}_i) \;\le\; \frac{\epsilon}{p} \;+\; \widehat{L}_{\text{probe}}(\theta^\star) \;+\; \frac{\lambda}{2}\,\|\theta^\star\|_F^2.$$

By Corollary B.26 applied at $W = \theta^\star$ and Lemma B.23,

$$\widehat{L}_{\text{probe}}(\theta^\star) \;\le\; \widehat{L}_{\text{true}}(\theta^\star) \;+\; R\,\|\theta^\star\|_F\,\Delta_{1,n} \;\le\; \epsilon \;+\; \Gamma_\star\sqrt{\frac{\log(3/\kappa)}{2n}} \;+\; R\,\|\theta^\star\|_F\,\Delta_{1,n}.$$

Hence

$$\widehat{L}_{\text{probe}}(\tilde{\theta}_i) \;\le\; \frac{\epsilon}{p} + \epsilon \;+\; \frac{\lambda}{2}\,\|\theta^\star\|_F^2 \;+\; \Gamma_\star\sqrt{\frac{\log(3/\kappa)}{2n}} \;+\; R\,\|\theta^\star\|_F\,\Delta_{1,n}.$$

Bound on $\widehat{L}_{\text{true}}(\tilde{\theta}_i)$. By Corollary B.26 at $W = \tilde{\theta}_i$,

$$\widehat{L}_{\text{true}}(\tilde{\theta}_i) \;\le\; \widehat{L}_{\text{probe}}(\tilde{\theta}_i) \;+\; R\,\|\tilde{\theta}_i\|_F\,\Delta_{1,n}.$$

Substituting the bound from the previous step,

$$\widehat{L}_{\text{true}}(\tilde{\theta}_i) \;\le\; \frac{\epsilon}{p} + \epsilon \;+\; \frac{\lambda}{2}\,\|\theta^\star\|_F^2 \;+\; \Gamma_\star\sqrt{\frac{\log(3/\kappa)}{2n}} \;+\; R\,(\|\theta^\star\|_F + \|\tilde{\theta}_i\|_F)\,\Delta_{1,n}.$$

On $\mathcal{E}_{\text{pinsker}}$, by Lemma B.22,

$$\Delta_{1,n} \;\le\; \sqrt{2B_{\text{CE}}} \;+\; \sqrt{\frac{\log(3/\kappa)}{2n}}.$$

Therefore,

$$\widehat{L}_{\text{true}}(\tilde{\theta}_i) \;\le\; \frac{\epsilon}{p} + \epsilon \;+\; \frac{\lambda}{2}\,\|\theta^\star\|_F^2 \;+\; R\,(\|\theta^\star\|_F + B_\theta)\,\sqrt{2B_{\text{CE}}} \;+\; \Big(\Gamma_\star + R\,(\|\theta^\star\|_F + B_\theta)\Big)\sqrt{\frac{\log(3/\kappa)}{2n}}.$$

Bound on $\mathcal{R}(\tilde{\theta}_i)$. By Lemma B.24, on $\mathcal{E}_{\text{u}}$ and since $\|\tilde{\theta}_i\| \le B_\theta$ by Lemma B.21,

$$\mathcal{R}(\tilde{\theta}_i) \;\le\; \widehat{L}_{\text{true}}(\tilde{\theta}_i) \;+\; 2\sqrt{2}\,RB_\theta\sqrt{\frac{K}{n}} \;+\; \Gamma_B\sqrt{\frac{\log(3/\kappa)}{2n}}.$$

Combining with the previous bound and substituting $B_\theta = \sqrt{2(1+p)\log K/(\lambda\,p)}$ (which yields

$2\sqrt{2}RB_\theta\sqrt{K/n} = 4R\sqrt{(1+p)(K\log K)/(\lambda\, p\, n)})$ gives the stated bound. □

Lemma B.20 (Stationarity bound)**.** *At a stationary point* $\tilde{\Theta}$*, for learner* $i \in U$*,*

$$p\,\widehat{L}_{\text{probe}}(\tilde{\theta}_i) \;\le\; \epsilon \;+\; p\,\widehat{L}_{\text{probe}}(\theta^\star) \;+\; \frac{\lambda\, p}{2}\,\|\theta^\star\|_F^2.$$

Proof of Lemma B.20. By optimality of $\tilde{\theta}_i$ for $\Phi_i(\cdot;\tilde{\Theta})$,

$$\Phi_i(\tilde{\theta}_i;\tilde{\Theta}) \;\le\; \Phi_i(\theta^\star;\tilde{\Theta}).$$

Subtracting the two population CE terms at $W = \tilde{\theta}_i$ and $W = \theta^\star$, and using that

$$\tau\alpha_i\;\mathbb{E}_{\mathscr{P}_i}\,\ell_{\text{CE}}(\theta^\star) \;+\; (1-\tau)\,a_i(\tilde{\Theta})\;\mathbb{E}_{\mathscr{D}_i(\tilde{\Theta})}\,\ell_{\text{CE}}(\theta^\star) \;\le\; \epsilon,$$

we obtain the stated bound. □

Lemma B.21 (Norm bound and probability floor)**.** *We have* $\|\tilde{\theta}_i\|_F \le B_\theta = \sqrt{2(1+p)\log K/(\lambda\, p)}$*. Consequently, for any* x*,*

$$\min_{c\in[K]} q_{\tilde{\theta}_i}(x)_c \;\ge\; \gamma_B \;=\; \frac{1}{K}\exp\bigl(-2RB_\theta\bigr), \quad \Gamma_B \;=\; \log\frac{1}{\gamma_B} \;=\; \log K + 2RB_\theta.$$

Proof of Lemma B.21. Compare $\Phi_i(\tilde{\theta}_i;\tilde{\Theta})$ to $\Phi_i(0;\tilde{\Theta})$. For $W = 0$, q_W is uniform and each CE term equals $\log K$. Thus

$$\frac{\lambda\, p}{2}\,\|\tilde{\theta}_i\|_F^2 \;\le\; (1+p)\,\log K \quad\Rightarrow\quad \|\tilde{\theta}_i\|_F \;\le\; \sqrt{\frac{2(1+p)\log K}{\lambda\, p}}.$$

For the floor, write for any c, c':

$$|z_c - z_{c'}| \;=\; |(W_c - W_{c'})^\top x| \;\le\; \|W_c - W_{c'}\|\,\|x\| \;\le\; 2\,\|W\|_F\,\|x\| \;\le\; 2R\|W\|_F.$$

Hence $q_W(x)_c \ge \frac{1}{K}\exp(-2R\|W\|_F)$. □

Lemma B.22 (Pseudo-label discrepancy)**.** *Let* $\Delta_{1,n} = \frac{1}{n}\sum_{q=1}^n \|\tilde{y}_i^q - y_i^q\|_1$*. Under Assumption B.1,*

for any $\kappa \in (0,1)$*, with probability at least* $1-\kappa/3$*,*

$$\Delta_{1,n} \;\le\; \sqrt{2B_{\mathrm{CE}}} \;+\; \sqrt{\frac{\log(3/\kappa)}{2n}}.$$

Proof of Lemma B.22. For one-hot y, $\mathrm{CE}(y,\tilde{y}) = \mathrm{KL}(y\|\tilde{y})$ and Pinsker gives $\mathbb{E}\,\|y-\tilde{y}\|_1 \le \sqrt{2\,\mathbb{E}\,\mathrm{KL}(y\|\tilde{y})} \le \sqrt{2B_{\mathrm{CE}}}$. Since $\|y-\tilde{y}\|_1 \in [0,2]$, Hoeffding yields the stated bound. □

Lemma B.23 (Concentration at $\theta^\star$)**.** *With probability at least* $1-\kappa/3$*,*

$$\left|\widehat{L}_{\mathrm{true}}(\theta^\star) - \epsilon\right| \;\le\; \Gamma_\star \sqrt{\frac{\log(3/\kappa)}{2n}},$$

where $\Gamma_\star = \log K + 2R\|\theta^\star\|_F$*.*

Proof of Lemma B.23. By Lemma B.21 applied to $W = \theta^\star$, $\min_c q_{\theta^\star}(x)_c \ge \gamma_\star$, hence $\ell_{\mathrm{CE}}((x,y),\theta^\star) \in [0,\Gamma_\star]$. Hoeffding's inequality gives the result. □

Lemma B.24 (Uniform convergence)**.** *With probability at least* $1-\kappa/3$*,*

$$\sup_{\|W\|_F \le B_\theta} \left|\widehat{L}_{\mathrm{true}}(W) - \mathcal{R}(W)\right| \;\le\; 2\sqrt{2}\,R\,B_\theta \sqrt{\frac{K}{n}} \;+\; \Gamma_B \sqrt{\frac{\log(3/\kappa)}{2n}}.$$

Proof of Lemma B.24. For any sample $\{(x_i,y_i)\}_{i=1}^n$ with $\|x_i\| \le R$, we bound the empirical Rademacher complexity of

$$\mathcal{F}_{B_\theta} \;:=\; \big\{(x,y) \mapsto \ell_{\mathrm{CE}}((x,y),W) : \|W\|_F \le B_\theta\big\}.$$

The function $f(z,y) = -\log \mathrm{softmax}(z)_Y$ has gradient $\nabla_z f = q - y$ with $\|\nabla_z f\|_2 = \|q-y\|_2 \le \sqrt{2}$. Thus f is $\sqrt{2}$-Lipschitz in z. By the vector contraction lemma,

$$\mathfrak{R}_n(\mathcal{F}_{B_\theta}) \;\le\; \sqrt{2}\cdot \mathbb{E}\left[\sup_{\|W\|_F \le B_\theta} \frac{1}{n}\sum_{i=1}^n \sum_{k=1}^K \sigma_{i,k}\,(Wx_i)_k\right],$$

with $\sigma_{i,k}$ i.i.d. Rademacher. The inner supremum equals

$$\frac{B_\theta}{n}\,\mathbb{E}\Big\|\sum_{i=1}^n\sum_{k=1}^K \sigma_{i,k}\,e_k x_i^\top\Big\|_F \;\le\; \frac{B_\theta}{n}\sqrt{\mathbb{E}\Big\|\sum_{i,k}\sigma_{i,k}\,e_k x_i^\top\Big\|_F^2} \;=\; \frac{B_\theta}{n}\sqrt{\sum_{k=1}^K\sum_{i=1}^n \|x_i\|^2} \;\le\; B_\theta R\sqrt{\frac{K}{n}}.$$

Hence $\mathfrak{R}_n(\mathcal{F}_{B_\theta}) \le \sqrt{2}\, B_\theta R\sqrt{K/n}$. A standard symmetrization and bounded-difference argument (the per-sample loss range over $\|W\| \le B_\theta$ is at most Γ_B by Lemma B.21) yields, with prob. $\ge 1-\kappa/3$,

$$\sup_{\|W\|_F \le B_\theta} \left|\widehat{L}_{\text{true}}(W) - \mathcal{R}(W)\right| \;\le\; 2\,\mathfrak{R}_n(\mathcal{F}_{B_\theta}) \;+\; \Gamma_B \sqrt{\frac{\log(3/\kappa)}{2n}},$$

which gives the stated bound. □

Lemma B.25 (Logits bridge for cross-entropy; no probability floors)**.** *Let $W \in \mathbb{R}^{K\times d}$, $z(x) = Wx \in \mathbb{R}^K$, and $q_W(x) = \text{softmax}(z(x))$. For any $x \in \mathbb{R}^d$, any one-hot label $y \in \{e_1, \dots, e_K\}$, and any soft pseudo-label $\tilde{y} \in \Delta^{K-1}$ (i.e., $\tilde{y}_c \ge 0$, $\sum_c \tilde{y}_c = 1$), the cross-entropy difference satisfies the exact identity*

$$\text{CE}(\tilde{y}, q_W) - \text{CE}(y, q_W) \;=\; -\langle \tilde{y} - y,\; z(x)\rangle.$$

Consequently,

$$\left|\,\text{CE}(\tilde{y}, q_W) - \text{CE}(y, q_W)\right| \;\le\; \|\tilde{y} - y\|_1 \cdot \|z(x)\|_\infty \;\le\; \|\tilde{y} - y\|_1 \cdot R\,\|W\|_F.$$

Proof. Recall that $\text{CE}(r, q) = -\sum_{c=1}^K r_c \log q_c$ for any probability vector r, and $q_c = \frac{e^{z_c}}{\sum_j e^{z_j}}$ with $z = Wx$. Hence

$$-\log q_c \;=\; \log\Big(\sum_{j=1}^K e^{z_j}\Big) - z_c \;=\; \text{logsumexp}(z) - z_c.$$

Therefore, for any $r \in \Delta^{K-1}$,

$$\text{CE}(r, q_W) \;=\; \sum_{c=1}^K r_c \left(\text{logsumexp}(z) - z_c\right) \;=\; \text{logsumexp}(z)\cdot \underbrace{\sum_{c=1}^K r_c}_{=1} \;-\; \sum_{c=1}^K r_c\, z_c \;=\; \text{logsumexp}(z) - \langle r, z\rangle.$$

Applying this twice with $r = \tilde{y}$ and $r = y$ gives

$$\text{CE}(\tilde{y}, q_W) - \text{CE}(y, q_W) \;=\; \big(\text{logsumexp}(z) - \langle \tilde{y}, z\rangle\big) \;-\; \big(\text{logsumexp}(z) - \langle y, z\rangle\big) \;=\; -\langle \tilde{y} - y, z\rangle,$$

which is the claimed identity.

For the inequalities, use Hölder and the fact that $\|z\|_\infty = \max_c |z_c|$:

$$\big|\,\mathrm{CE}(\tilde{y}, q_W) - \mathrm{CE}(y, q_W)\big| \;=\; |\langle \tilde{y} - y, z\rangle| \;\le\; \|\tilde{y} - y\|_1 \, \|z\|_\infty.$$

Finally, since $z_c = W_c^\top x$ and $\|x\| \le R$,

$$\|z\|_\infty \;=\; \max_c |W_c^\top x| \;\le\; \max_c \|W_c\|_2 \cdot \|x\| \;\le\; \Big(\max_c \|W_c\|_2\Big)\, R \;\le\; R\,\|W\|_F,$$

because $\|W\|_F^2 = \sum_c \|W_c\|_2^2 \ge \max_c \|W_c\|_2^2$. Combining the bounds yields the result. □

Lemma B.26 (Empirical bridge on the probing batch)**.** *On the probing dataset* $\{(\tilde{x}_i^q, \tilde{y}_i^q, y_i^q)\}_{q=1}^n$, *define*

$$\widehat{L}_{\mathrm{probe}}(W) \;=\; \frac{1}{n}\sum_{q=1}^n \mathrm{CE}(\tilde{y}_i^q, q_W(\tilde{x}_i^q)), \quad \widehat{L}_{\mathrm{true}}(W) \;=\; \frac{1}{n}\sum_{q=1}^n \mathrm{CE}(y_i^q, q_W(\tilde{x}_i^q)), \quad \Delta_{1,n} \;=\; \frac{1}{n}\sum_{q=1}^n \|\tilde{y}_i^q - y_i^q\|_1.$$

Then for any W,

$$\big|\widehat{L}_{\mathrm{probe}}(W) - \widehat{L}_{\mathrm{true}}(W)\big| \;\le\; R\,\|W\|_F \cdot \Delta_{1,n}.$$

Proof. Apply Lemma B.25 termwise and average. □