LEARNING TO ASSIGN PREDICTION TASKS TO AGENTS WITH CAPACITY CONSTRAINTS

Anonymous authors
Paper under double-blind review

ABSTRACT

We address the problem of learning to assign prediction tasks to one agent from a set of available agents, including human decision-makers and AI models. We focus on sequential learning of agent expertise and assignment policies where each agent is constrained to handle a fraction of tasks. We provide a general theoretical characterization of this problem in terms of agent capacities, differences in agent expertise, and task context. We then develop a framework of sequential explore-exploit policy-learning algorithms that seek to maximize overall performance. Experimental results over a variety of tabular, image, and text prediction tasks demonstrate systematic gains from our policy-learning algorithms relative to non-contextual baselines across different types of agents, including LLMs and humans.

1 INTRODUCTION

In many real-world settings, tasks arrive sequentially and need to be allocated to agents, where the agents (AI or human) may differ in their expertise. An example is in the context of medical radiology, where the set of agents could include an AI model and multiple human radiologists with different experiences, each available for tasks such as providing a diagnostic class label for a radiology image. For many types of tasks, human expertise is often quite specific (Ericsson et al., 1993; Chase & Simon, 1973). For example, clinicians tend to perform well on familiar types of medical cases but do not generalize equally across all cases (Eva, 2005; Norman et al., 2007). AI models also can differ in terms of their expertise, for example, in large language model (LLM) routing where the focus is on determining which LLM is best-suited to answering different queries (Srivatsa et al., 2024; Song et al., 2025). Further, in the context both human and AI expertise, the value of human-AI complementarity is widely acknowledged (Gonzalez et al., 2026), across applications such as medical diagnosis (Zöller et al., 2025), image classification (Steyvers et al., 2022), and content moderation (Lai et al., 2022). These findings suggest that different types of prediction agents can have localized expertise, and that effective task assignment should account for heterogeneous, context-dependent performance.

Another component of many real-world task assignment problems is the presence of **capacity constraints** on how much workload an individual agent can handle, i.e., what fraction of tasks in the long-run they are assigned. These are distinct from cost constraints: capacity constraints impose agent-specific workload limits, rather than only limiting aggregate spending. For example, in the radiology example earlier, assigning many or all tasks to the most senior/experienced human radiologist is typically not a practical option (e.g., due to human fatigue or burnout (Berlin, 2000; Alzoubi et al., 2024)). Similarly, assigning all tasks to an AI agent (and none to humans) may be sub-optimal in terms of leveraging relevant human expertise and may be undesirable from a safety and employee satisfaction (Salikutluk et al., 2024; Zhang et al., 2022; Shneiderman, 2020).

Our work differs from prior work in two primary aspects. First, we jointly learn assignment policies and agent expertise online (useful, for example, under distribution shift), unlike prior work on learning to defer or LLM routing, which learns agent expertise offline (Mozannar & Sontag, 2020; Claure et al., 2020; Chen et al., 2023). Second, we study this problem under capacity constraints on the agents, unlike fairness-constrained bandits, which enforce a shared minimum-rate constraint across arms (Chen et al., 2020), and cost-constrained LLM routing, which enforces a global budget rather than agent-specific capacity limits (Panda et al., 2025; Li et al., 2026; Taberner-Miller, 2026). Our technical approach builds on contextual multi-armed bandit frameworks, with tasks arriving sequentially and the assignment policy and agent expertise models being updated from observed rewards. Across

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

simulated agents, real human-annotation data, machine learning classifiers, and LLMs, we show how contextual assignment outperforms non-contextual assignment.[1] More broadly, our framework highlights the value of jointly modeling latent expertise and capacity constraints in sequential decision-making environments, such as human-AI collaboration and large-scale organizational systems. Our work makes the following contributions:

- We provide a theoretical framework for capacity-constrained task assignment, introducing for example the concept of a *contextual reward gap* between agents as a general framework for providing insights into task allocation and for characterizing optimal policies.
- We conduct a series of experiments across settings with human annotators, machine learning classifiers, and LLMs, including distribution shift, showing systematic gains for contextual capacity-constrained policies over non-contextual baselines.[2]

2 Optimal Assignment Policies and Capacity Constraints

Below we define the assignment problem and characterize the structure of optimal policies in an oracle setting where agents' context-dependent expertise is assumed known. This allows us to isolate how contextual assignment and capacity constraints shape the optimal allocation rule. Later in Section 3, we address the practical problem of online learning of expertise.

2.1 Tasks, Agents, and Rewards

There are A prediction agents indexed by $a \in \{1, \ldots, A\}$, where agents can be machine learning models or humans. We assume that each prediction task needs to be assigned to a single agent to make a prediction. Prediction tasks arrive sequentially, where each task is characterized by an observable **context** $x_t \in \mathcal{X} \subseteq \mathbb{R}^d$, $t = 1, \ldots, T$, and contexts are drawn i.i.d. from some unknown distribution P_X. If assigned a task, a prediction agent a (a model or a human) generates a prediction $\hat{y}_{a,t}$. In what follows, we will focus on classification tasks and agents, where $\hat{y}_{a,t}$ is from a set of K possible labels, but the framework we discuss can be directly extended to other prediction problems such as regression. The agent's prediction $\hat{y}_{a,t}$ can be based not only on the context x_t but on features beyond the context (e.g., an agent could use all pixels in an image as input, whereas the context could be some lower-dimensional representation of the pixel information).

Assigning task t to agent a yields a stochastic reward $r_{t,a} \sim P_a(\cdot \mid x_t)$, where $P_a(\cdot \mid x)$ denotes the agent-specific reward distribution conditioned on context x_t. Rewards reflect the prediction quality of the agent. A natural choice for reward in a classification context (and one we will focus on in this paper) is the accuracy of agent a's prediction for task t, i.e., $r_{t,a} = \mathbb{1}\{\hat{y}_{a,t} = y_t\}$. Here, y_t is the true observed label for the task, observed after a prediction is made[3]. Alternative reward definitions (not explored here) could be negative log-loss or negative squared error (Brier score).

We define the **conditional expected reward** of agent a as $\mu_a(x) := \mathbb{E}[r_{t,a} \mid x_t = x]$, which captures agent a's performance conditioned on context x, with the corresponding **marginal reward** defined as $\mu_a := \mathbb{E}_{x \sim P_X}[\mu_a(x)]$, representing agent a's overall performance with respect to the task/context distribution. Using accuracy (as above) as the definition of reward, these expectations correspond to conditional expected accuracy as a function of context x and marginal accuracy. To illustrate these ideas, Figure 1 shows a simple case of two LLM classifier agents. The x-axis represents context x and the y-axis corresponds to the estimated empirical conditional accuracy for each agent, with varying agent expertise (accuracy, or expected reward) as a function of context.

2.2 Optimal Assignment Policies without Agent Constraints

Definition 2.1 (Assignment Policy)**.** We define an **assignment policy** ϕ as a (possibly stochastic) mapping from a task x to an agent $a \in \{1, \ldots, A\}$. Let $P_\phi(a \mid x)$ denote the probability that policy

[1] In this paper, we use "non-contextual assignment" and "random assignment" interchangeably.

[2] We will release the code, experiment scripts, and datasets in the final version of the paper.

[3] While we assume immediate feedback in our main setting, the framework naturally extends to settings with delayed rewards, as standard contextual bandit methods can accommodate delayed updates (Blanchet et al., 2024).

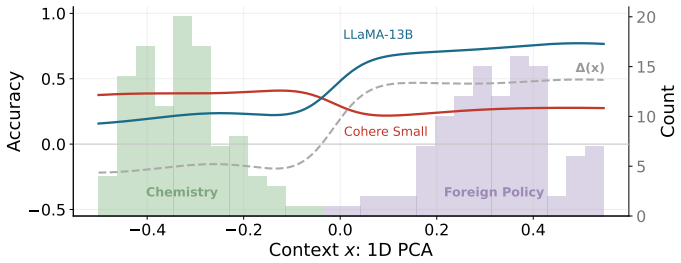


Figure 1: Accuracy (smoothed, empirically estimated) of two LLMs as a function of context. Context is defined as the first principal component of embeddings of multiple-choice questions from MMLU (Hendrycks et al., 2020), with embeddings from `sentence-transformers/all-MiniLM-L6-v2`. Cohere Small performs better on College Chemistry tasks (left), while LLaMa-13B performs better on US Foreign Policy tasks (right). The dashed gray curve denotes the estimated contextual reward gap $\Delta(x) := \mu_{\text{LLaMa}}(x) - \mu_{\text{Cohere}}(x)$. The marginal accuracies are 0.32 for Cohere Small and 0.47 for LLaMa-13B.

ϕ assigns a task to agent a given context x. For deterministic policies we have $P_\phi(a \mid x) = 1$ for a single agent $a = \phi(x)$ and 0 otherwise. The **marginal reward** of a policy ϕ is

$$\mu^\phi := \mathbb{E}_{x\sim P_X}\left[\sum_{a=1}^{A} P_\phi(a \mid x)\mu_a(x)\right].$$

As a baseline, consider a non-contextual (random) policy that ignores x and assigns tasks using some fixed set of probabilities $\{P_a\}$, where $\sum_a P_a = 1$, resulting in $\mu^\phi = \sum_{a=1}^{A} P_a\mu_a$. Similarly, a policy that always assigns tasks to a single fixed agent a' (such as the agent with the highest marginal reward) corresponds to $P_{a'} = 1$, resulting in marginal reward $\mu^\phi = \mu_{a'}$. For context-dependent deterministic policies, ϕ can be represented via scoring functions $\{s_a(x)\}_{a=1}^{A}$ such that $\phi(x) = \arg\max_a\{s_a(x)\}$. This partitions the context space into regions $\mathcal{R}_a = \{x : \phi(x) = a\}$. The marginal reward averages over the context distribution P_X, so $\mu^\phi = \sum_{a=1}^{A}\int_{\mathcal{R}_a} \mu_a(x)\, P_X(x)\, dx$.

Definition 2.2. The **optimal (unconstrained) policy** is $\phi^\star(x) = \arg\max_a \mu_a(x)$, i.e., it assigns each context x to the agent with the highest conditional expected reward for that context. Equivalently, the optimal decision regions $\{\mathcal{R}_a^\star\}$ satisfy $\mathcal{R}_a^\star = \{x \in \mathcal{X} : \phi^\star(x) = a\}$.

Two-agent case without constraints: As mentioned earlier, Figure 1 illustrates heterogeneous expertise: different agents perform better in different regions of the context space, indicating that the optimal policy requires assigning tasks based on context. To formalize this, consider two agents with context-dependent performance $\mu_1(x)$ and $\mu_2(x)$, and define the **contextual reward gap** $\Delta(x) := \mu_1(x) - \mu_2(x)$; see the dashed gray curve in Figure 1 for an empirical estimate. The optimal (unconstrained) policy assigns each task to the agent with higher expected reward, $\phi^\star(x) = \arg\max_{a\in\{1,2\}} \mu_a(x)$, equivalent to a threshold rule in $\Delta(x)$. In contrast, a context-independent policy selects a single agent based on marginal performance, yielding reward $\max\{\mu_1, \mu_2\}$, where $\mu_a = \mathbb{E}_x[\mu_a(x)]$. This policy ignores variation in $\Delta(x)$ and is generally suboptimal.

Gains by a context-dependent policy arise whenever $P_X(\Delta(x) > 0) > 0$ and $P_X(\Delta(x) < 0) > 0$ (see Appendix B). A special case arises when one agent dominates across all contexts, i.e., $\mu_{a^\star}(x) \geq \mu_a(x)$ for all a and x, in which case assigning all tasks to $a^\star$ is optimal. For context-dependent policies, the performance gap over a non-contextual policy is driven by regions where $\Delta(x)$ has the same sign, i.e., regions in the context space where one agent strictly dominates the other.

2.3 Optimal Assignment Policies with Agent Capacity Constraints

We now extend the framework to incorporate long-run capacity constraints across A agents, meaning that each agent's capacity target is enforced in expectation over time. Let $\alpha_a \in [0, 1]$ denote the target long-run fraction of tasks assigned to agent a, with $\sum_{a=1}^{A} \alpha_a = 1$. The objective is to maximize expected reward subject to these long-run capacity constraints. We assume that the capacity constraints α_a are known and fixed for the problem, e.g., a company or hospital might have

a policy that 90% of tasks are assigned to a machine learning model and 10% to a human expert. This notion of capacity differs from the cost-budget constraints studied in Panda et al. (2025): a cheaper, better-performing agent cannot simply be used for all tasks, because each agent has its own long-run assignment target. For a non-contextual assignment policy, the only option is to assign tasks (randomly) based on probabilities. $P_a = \alpha_a$, yielding $\mu^{\text{rand}} = \sum_{a=1}^{A} \alpha_a \mu_a$. For the theoretical characterization below, we assume a continuous context distribution: deterministic threshold policies achieve the target assignment capacities exactly. With discrete contexts, the same characterization applies, where exact equality requires stochastic tie-breaking at boundary contexts.

Definition 2.3. The **optimal constrained policy** maximizes $\mu^{\phi} = \mathbb{E}_x\Big[\sum_{a=1}^{A} P_\phi(a \mid x)\mu_a(x)\Big]$ subject to capacity constraint $\mathbb{E}_x[P_\phi(a \mid x)] = \alpha_a$, $a = 1, \ldots, A$, $A \geq 2$.

Theorem 2.1 (General Form of Optimal Constrained Policy)**.** *The optimal policy takes a simple form: there exist constants* $\{\lambda_a\}_{a=1}^{A}$ *such that*

$$\phi^\star(x) = \arg \max_{a \in \{1,\ldots,A\}} \big\{\mu_a(x) - \lambda_a\big\}.$$

See Appendix C for proof and additional details. In the optimal policy, each task is assigned to the agent with the highest adjusted reward (i.e., $\mu_a(x) - \lambda_a$), where λ_a acts as a shadow price for capacity. This policy is at least as good as the non-contextual assignment (i.e., $\mu^{\phi^\star} \geq \mu^{\text{rand}}$), with strict inequality whenever the identity of the best agent depends on the context x. In practice, both the reward functions $\mu_a(x)$ and the shadow prices $\{\lambda_a\}$ are unknown. In Section 3, we introduce a queue-based method that jointly learns $\mu_a(x)$ and approximates λ_a, $a = 1, \ldots, A$.

Two-agent case with capacity constraints. We consider a simplified setting with two agents, with capacities $\alpha_1 = \alpha$ and $\alpha_2 = 1 - \alpha$, with contextual reward gap $\Delta(x) = \mu_1(x) - \mu_2(x)$.

Proposition 2.1 (Two-Agent Optimal Constrained Policy)**.** *Under capacity constraints, the optimal policy assigns agent* 1 *to the* α*-mass of contexts with the largest values of* $\Delta(x)$*, and agent* 2 *to the remainder, yielding a threshold rule in* $\Delta(x)$*. See Appendix D.1 for details.*

In the non-dominant case, where both $P_X(\Delta(x) > 0) > 0$ and $P_X(\Delta(x) < 0) > 0$, agents outperform each other in different regions of the context space. For example, Figure 1 shows that the two LLMs specialize in different regions of context space, so $\Delta(x)$ changes sign. In this example, if LLaMa-13B has 25% capacity and Cohere Small has 75%, then the optimal capacity-constrained policy would intuitively threshold the context around $x = 0.3$. We also note that the combined agents' performance can outperform either individual agent under certain conditions. In particular, when $\alpha = P_X(\Delta(x) \geq 0)$, the policy assigns each context to the better-performing agent, and the resulting reward satisfies $\mu^{\phi^\star} > \max\{\mu_1, \mu_2\}$ (see Appendix D.3).

More generally, the gains from contextual assignment are driven by disagreement between agents. Let $\mathcal{D}_1 = \{x : \Delta(x) > 0\}$ and $\mathcal{D}_2 = \{x : \Delta(x) < 0\}$ denote the regions where each agent has an advantage. The improvement over non-contextual assignment depends on both the size of these **disagreement regions** and the magnitude of $\Delta(x)$, as well as the capacity α.

Proposition 2.2 (Full Exploitation of Disagreement)**.** *When capacity allows full exploitation of disagreement, i.e.,* $p_1 \leq \alpha \leq 1 - p_2$*, the optimal policy assigns all* $x \in \mathcal{D}_1$ *to agent* 1 *and all* $x \in \mathcal{D}_2$ *to agent* 2*.*

In this case, the gain over non-contextual assignment is $\mu^{\phi^\star} - \mu^{\text{rand}} = (1 - \alpha)p_1\mathbb{E}\big[\Delta(X) \mid X \in \mathcal{D}_1\big] + \alpha p_2\mathbb{E}\big[-\Delta(X) \mid X \in \mathcal{D}_2\big]$, where $p_1 = P_X(\mathcal{D}_1)$ and $p_2 = P_X(\mathcal{D}_2)$. Thus, the magnitude of improvement depends not only on the size of the disagreement regions but also on the average contextual reward gap within those regions. This formalizes and generalizes the disparity intuition in Chen et al. (2020): gains arise when agents have different relative advantages across contexts, and our characterization links those gains to the size of the disagreement regions and the capacity α. See Appendix D.4 for details.

Proposition 2.3 (Optimal Policy under Agent Dominance)**.** *When one agent is dominant, i.e.,* $\mu_1(x) \geq \mu_2(x)$ *for all* x*, the optimal policy assigns agent* 1 *to the* α*-mass of contexts with the largest values of* $\Delta(x)$*. As* $\alpha \to 1$*, the selected region* $\mathcal{R}_1^\star$ *expands to the entire context space* $\mathcal{X}$*, and the resulting reward satisfies* $\mu^{\phi^\star} \to \mu_1$*, converging to the unconstrained optimum.*

Since $\mu_1(x) \geq \mu_2(x)$ for all x, we have $\Delta(x) \geq 0$ everywhere. Because $\Delta(x)$ is nonnegative for all x, the threshold characterization corresponds to selecting the contexts where agent 1 has the greatest advantage over agent 2. As a specialization to the two-agent setting, the contextual policy satisfies $\mu^{\phi^\star} \geq \mu^{\text{rand}}$ in both the non-dominant and dominant cases, with strict inequality whenever agent performance depends on the context (see Appendix D.2).

3 Online Learning of Assignment Policies with Agent Capacities

Below we develop a Bayesian multi-arm bandit (MAB) framework for learning the optimal assignment policy online when the conditional expected rewards $\mu_a(x)$ are unknown. At each round t, a context x_t arrives, an agent a_t is selected by the current policy (taking constraints into account), a reward r_{t,a_t} is observed, and the contextual reward model is updated for the selected agent.

3.1 Context Modeling

For context models $\mu_a(x)$, we consider logistic and tree-based models as described below. For the **logistic context model**, agent a's conditional expected reward is modeled via $\mu_a(x_t) = \mathbb{P}(r_{t,a} = 1 \mid x_t) = \sigma(\theta_a^\top x_t)$, where θ_a is a latent parameter vector governing agent a's context-dependent expertise and $\sigma(z) = (1 + e^{-z})^{-1}$. Following Chapelle & Li (2011), we maintain an approximate Gaussian posterior for θ_a over rounds t using a Laplace approximation. We investigate two MAB strategies in this context (alternatives such as the upper-confidence bound (UCB) method could also be used). The first is Thompson sampling (TS), where at each round t, a parameter vector is sampled from the current posterior distribution over θ_a, the corresponding sampled reward estimate $\tilde{\mu}_{a,t}(x_t) = \sigma(\tilde{\theta}_{a,t}^\top x_t)$ is computed, and this value is used by the assignment policy to select an agent. The second strategy is greedy, using an MAP point estimate $\hat{\theta}_{a,t}$ to compute $\hat{\mu}_{a,t}(x_t) = \sigma(\hat{\theta}_{a,t}^\top x_t)$. The greedy policy selects agents based on this estimated expected reward, without explicit posterior sampling. See details in Appendix E.1

To capture non-linear dependencies in $\mu_a(x_t)$, we also use **tree-based (random forest) context models**, where $\mu_a(x_t)$ is estimated by averaging predictions across an ensemble of trees trained on a set of B bootstrap samples. Again we investigate both greedy selection and Thompson sampling for assignment. For greedy selection, we use the ensemble mean $\mu_{a,t}(x_t) = \frac{1}{B}\sum_{b=1}^{B} f_{a,t}^{(b)}(x_t)$, where $f_{a,t}^{(b)}$ denotes the b-th tree for agent a at time t. To enable exploration, we approximate Thompson sampling using bootstrap uncertainty (similar to (Eckles & Kaptein, 2014)): at each round t, we sample a tree uniformly at random and use its prediction $\tilde{\mu}_{a,t}(x_t) = f_{a,t}^{(b)}(x_t), b \sim \text{Uniform}\{1, \ldots, B\}$. Additional details are in Appendix E.2.

3.2 Enforcing Capacity Constraints

To enforce the constraints online, we adopt a queue-based approach following Neely (2010); similar queue-based ideas have appeared in prior work on bandits with fairness constraints (Li et al., 2019; Huang et al., 2020) and human-AI service systems (Montanari et al., 2026). For each agent a, we maintain a virtual queue $Q_{t,a}$ that tracks deviations between realized assignments and the target capacity α_a. The queues evolve as

$$Q_{t+1,a} = \left[Q_{t,a} + I_{t,a} - \alpha_a\right]_+, \quad a = 1, \ldots, A, \tag{1}$$

where $I_{t,a} := \mathbb{1}\{a_t = a\}$ indicates whether agent a is selected at round t, and $[\cdot]_+ = \max(\cdot, 0)$. Intuitively, $Q_{t,a}$ increases when agent a is over-assigned relative to its target capacity and remains small otherwise. At each round t, the assignment balances estimated reward and capacity pressure:

$$a_t = \arg\max_{a \in \{1,\ldots,A\}} \{s_a(x_t)\}, \qquad \text{where } s_a(x_t) = \mu_{a,t}(x_t) - \eta Q_{t,a}, \tag{2}$$

where $\eta \geq 0$ controls the strength of the capacity penalty and $\mu_{a,t}(x_t)$ is generated either by TS or the greedy procedure for each agent. This rule can be interpreted as a dynamic approximation to the optimal policy of the form $\mu_a(x) - \lambda_a$, where $\{\lambda_a\}$ are the shadow prices arising from the constrained optimization problem (see Appendix C). $Q_{t,a}$ serves as a time-varying estimate of the shadow price. Intuitively, λ_a acts as a penalty for assigning additional tasks to agent a, discouraging

overuse when capacity is limited. When all agents have positive target capacities, this balancing mechanism also ensures that each agent continues to receive assignments and feedback, providing a form of implicit exploration even under greedy reward estimates. Our general online learning framework, which combines contextual reward models with queue-based capacity constraints, is summarized in Algorithm 1 in Appendix E. Implementation details are provided in Appendix F. Appendix G gives a finite-time bound on regret against the constrained optimum of Theorem 2.1, equal to ηT plus the regret of the underlying contextual bandit learner, together with a deterministic $O(A/\eta)$ bound on the virtual queues.

4 Experimental Results

We evaluate our framework across settings involving different types of classification agents for K-ary classification tasks with binary rewards. Our results demonstrate that contextual policies systematically outperform non-contextual baselines and individual agents across agent types and data modalities and also illustrate that capacity constraints induce enough implicit exploration for greedy policies to match Thompson Sampling. We consider machine learning classifiers, LLMs, and both human and simulated human agents. Simulated agents allow for controlled evaluation of heterogeneous expertise and learning dynamics (Madras et al., 2018; Mozannar & Sontag, 2020; Alves et al., 2024). Full details on experimental methods and datasets are provided in the Appendix. We focus here on the two-agent case ($A = 2$) for ease of visualization; multi-agent results are provided in Appendix H.5.

Across experiments, we induce distribution shift by training and evaluating on different data subsets, reflecting realistic deployment settings. Unless otherwise stated, all experiments follow the same evaluation protocol. We run each policy over 100 randomized permutations of the online task sequence and report average error rates. We vary the capacity of Agent 1 from 0 to 1, with Agent 2 taking the remaining capacity, and set the queue penalty parameter to $\eta = 0.5$. This value provides a practical trade-off between reward maximization and capacity enforcement; Appendix H.8 reports sensitivity analyses for other values of η. The main results in Figures 2b and 3 are averaged over 100 randomized task streams. For readability, error bars are only reported in Appendix H.3: across all datasets, contextual policies, and evaluated capacities, the standard errors of the mean error rates are small relative to the corresponding improvements over the non-contextual benchmark.

For reference, we consider a non-contextual baseline policy that randomly assigns each task to agent a with fixed probability α_a, independent of x, and thus does not exploit heterogeneity in $\mu_a(x)$. This baseline is also asymptotically equivalent in expected reward to learned non-contextual policies that estimate only marginal agent accuracy, such as $a_t \in \arg\max_a \{\hat{\mu}_{a,t} - \eta Q_{t,a}\}$ (details in Appendix D.5), because such policies do not condition assignments on x. This comparison in effect also provides a component ablation: replacing the contextual reward model with marginal reward estimates, while retaining the same queue-based capacity controller, yields the non-contextual baseline asymptotically, isolating the contribution of contextual expertise modeling. We note that existing algorithms used in the fair-bandit literature (Li et al., 2019; Huang et al., 2020; Chen et al., 2020; Claure et al., 2020) are not comparable as baselines since they address different problems than ours, e.g., where there is a common minimum rate of workload for all agents and workloads are unconstrained above that, and/or the algorithms are non-contextual (resulting in identical performance to our non-contextual algorithm implementation).

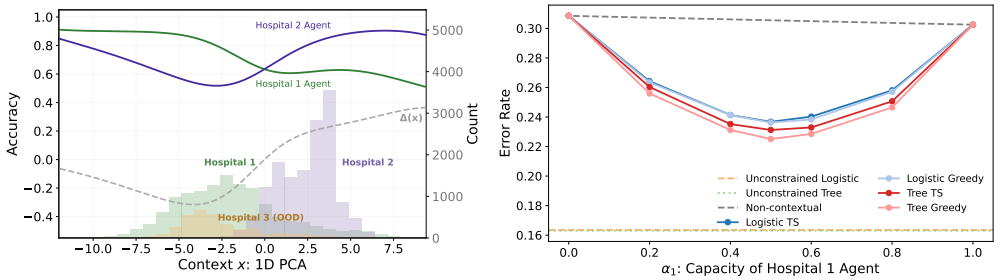
To assess how online and offline policy learning methods compare in terms of performance, we also consider an offline unconstrained benchmark that learns each agent's expertise using all available test data, and revisits this test data to assign tasks via $\phi(x_t) = \arg\max_a \hat{\mu}_a(x_t)$. This is an optimistic estimate of the error rate when agent expertise is learned offline without capacity constraints.

4.1 Experiments with Image, Tabular, and Text Classification Tasks

We start by introducing an example with the **Camelyon17** medical image classification dataset (Bandi et al., 2018). For illustration, we train 2 linear classifiers, each trained on entirely different data from 2 different hospitals, and then evaluated on a third hospital. We use the first principal component of several image features as context for the contextual policies (see Appendix H.1 for details). Figure 2a visualizes heterogeneity in agent expertise. The two agents exhibit different

accuracy profiles as a function of context, and the estimated contextual reward gap $\Delta(x)$ changes across the context space. This indicates that neither agent is uniformly best and that there is potential value in context-dependent assignment.

Figure 2b reports the average error rate as a function of Agent 1's capacity, where Agent 1 is trained on Hospital 1 data. The upper dotted line is the error rate for the non-contextual (random) policy with capacity α_1 (x-axis), where the two endpoints correspond to assigning all tasks to a single agent (resulting in the marginal error rates of the two agents). The lower dotted lines are the optimistic error rates from the offline unconstrained context-dependent models (logistic and tree-based, as described above). The intermediate lines are the error rates (averaged over runs) obtained from running the greedy and Thompson contextual policies for each of the logistic and tree-based context models, at different specific capacities $\alpha_1 \in \{0.2, 0.4, 0.5, 0.6, 0.8\}$. All contextual policies outperform the non-contextual baseline across all capacity levels. Moreover, the combined policy can achieve a significantly lower error than the non-contextual policy (e.g., at $\alpha_1 = 0.5$), demonstrating that leveraging context-dependent expertise systematically improves performance, and can result in performance better than either agent's marginal error rate, consistent with the theoretical results in Section 2.3. The offline unconstrained policies assign roughly half of the tasks (not shown on the graph) to Agent 1 for both logistic and tree models, indicating that the unconstrained context-dependent allocation is close to $0.5/0.5$ split. Thus, $\alpha_1 = 0.5$ is the empirical capacity allocation closest to allowing full exploitation of disagreement, in the sense of Proposition 2.2. The "error gap" between the optimistic offline error of 0.16 and the online policies' average error of roughly 0.23 at $\alpha_1 = 0.5$ reflects the cost of online learning, while moving away from this point introduces an additional error penalty from enforcing the constraints.

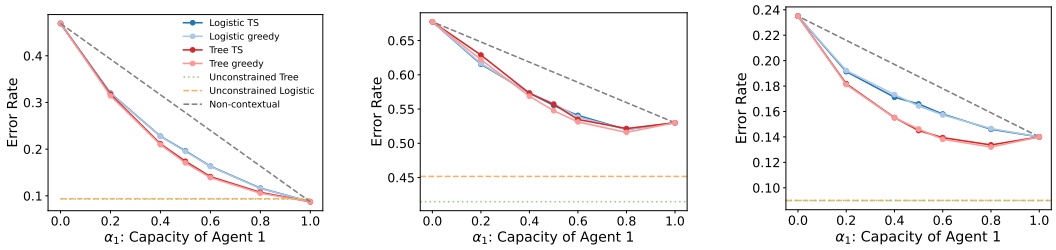


(a) (b)

Figure 2: (a) Empirical accuracy as a function of context, (b) error rate vs capacity of Agent 1 averaged over 2000 image classification tasks and 100 random online streams from the **Camelyon17**. Uncertainty estimates are reported in Appendix H.3.

We further evaluate our framework on additional tabular, text, and image datasets, with the main text reporting one representative dataset of each type and Appendix H.2 reporting the remaining four. For the tabular dataset, we construct two agents with heterogeneous expertise: a logistic regression model trained on a random subset of up to 20% of the features, and an XGBoost classifier trained on the remaining features. Appendix H.10 shows that the gain over random assignment is not specific to this 20/80 split. The contextual policy observes the full feature space. Each agent is trained on 70% of the data, and the remaining 30% is used to simulate the online task sequence. We report results here on **Bank** (15 features, 12,357 tasks, predicting term deposit subscription). For the text-based setting, we evaluate pretrained LLM agents on **MMLU** (Hendrycks et al., 2020), using the models illustrated in Figure 1. Unlike the controlled feature-split construction used for the tabular datasets, these text-based settings use independently pretrained agents whose heterogeneous performance is not induced through feature partitioning. For the image dataset, we use the **ImageNet16H** dataset (Steyvers et al., 2022), which contains human classifications of noisy images. In this setting, the two agents are one human annotator and a fine-tuned VGG-19 convolutional neural network.

Figure 3 reports the average error rate as the capacity α_1 of Agent 1 (x-axis) varies from 0 to 1. As in Figure 2b, the endpoints correspond to the marginal error rates of the individual agents. Across these datasets, contextual policies (logistic and tree-based, both Thompson Sampling and Greedy) consistently outperform the non-contextual random baseline, highlighting the value of learning context-dependent expertise; Appendix H.2 shows the same pattern on four further datasets. The



(a) Bank (b) MMLU (c) ImageNet16H

Figure 3: Error rate as a function of capacity (two-agent case), for policies across datasets. Points represent the average error rate over 100 runs. Uncertainty estimates are reported in Appendix H.3.

combined policies do not necessarily outperform both individual agents, as performance depends on the distribution of $\Delta(x)$ and the capacity constraint (as discussed in Section 2.3). Multi-agent experiments show the same pattern, with contextual assignment improving performance (Appendix H.5). Tree-based models generally outperform logistic models, indicating the importance of capturing nonlinear and heterogeneous patterns in $\mu_a(x)$. Greedy and Thompson Sampling perform similarly across these experiments, holding the contextual reward model and capacity mechanism fixed. The similarity in performance is likely a result of the implicit exploration induced by the capacity constraints: each positive-capacity agent continues to receive assignments and feedback, letting greedy policies learn agent expertise over time (see also (Bayati et al., 2020) on the effectiveness of greedy contextual bandits). See Appendix H.1 for implementation details and H.4 for additional results. Appendix H.6 presents the case in which one agent (e.g., the AI model) has no capacity constraint. Appendix H.9 validates that realized assignments closely track the target capacities across datasets.

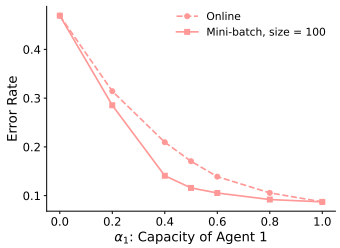
4.2 Mini-Batch Setting

An alternative to fully online assignment, is a mini-batch setting where assignments are made for N_B tasks at a time, and outcomes are observed after the batch is completed. This setting has an inherent latency-error tradeoff: batching can improve allocation under capacity constraints by considering multiple tasks jointly, but assignments and predictions are delayed. We illustrate this using the **Bank** dataset with the same agents as in Section 4.1. Figure 4a compares online and mini-batch performance with batch size $N_B = 100$ (50 runs) for the tree-based greedy policy. Mini-batch assignment yields modest but consistent improvements, as batching enables more efficient allocation by considering tasks jointly. We further vary the batch size from small values to the full dataset at a fixed capacity, $\alpha_1 = \alpha_2 = 0.5$, averaging results over 10 runs for each batch size (Figure 4b). The results exhibit a U-shaped error curve: small batches allow frequent updates but lead to noisy capacity enforcement, while large batches stabilize allocation but delay learning. Intermediate batch sizes balance these effects and achieve the lowest error. Overall, mini-batch assignment can improve performance over the fully online setting when a reasonable batch size is chosen, but this improvement comes at the cost of increased latency. See Appendix H.7 for implementation details and additional results.

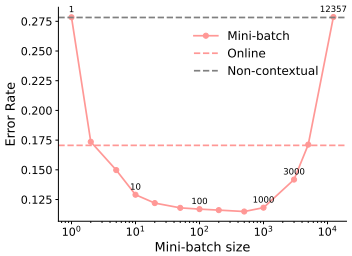
5 Related Work

Our work relates broadly to different strands of prior work in task routing and assignment problems. One such line of work is **learning to defer**: most work in this area studies a two-agent setting, with one model and one human, and typically learns the deferral rule offline from labeled data and observed expert decisions (Strong et al., 2025; Madras et al., 2018; Mozannar & Sontag, 2020; Mozannar et al., 2023; Alves et al., 2024). Recent papers incorporate workload and resource constraints (Keswani et al., 2021; Nguyen et al., 2025; Zhang et al., 2026; DeSalvo et al., 2025), but remain largely offline. Reid et al. (2024) studied online decision deferral, but assume full-information feedback, observing outcomes for both the model and the human regardless of the routing decision.

Another related line of work studies **LLM routing**, where an algorithm learns to route tasks among models with different costs and accuracies. Most existing work focuses on learning routing rules from offline evaluations or preference data to balance cost and accuracy (Chen et al., 2023; Ong et al.,



(a) Online vs mini-batch assignment with batch size $N_B = 100$. Error rates are averaged over 50 runs as the capacity of Agent 1 varies.



(b) Effect of batch size when Agents 1 and 2 have equal capacities, $\alpha_1 = \alpha_2 = 0.5$. Average error rates are computed over 10 runs.

Figure 4: Results of mini-batch assignment for the **Bank** dataset using the tree-based greedy policy.

2024), which differs from our approach, where we focus on online learning of agent abilities while respecting capacity constraints. More recent work has, however, studied online LLM routing under budget constraints (Panda et al., 2025; Li et al., 2026; Taberner-Miller, 2026), where the constraints can be global cost budgets, so that (e.g.) all tasks could be routed to a cheap and high-performing model as long as the total budget is satisfied. This differs from our setting, where agents face long-run capacity limits that capture operational constraints such as availability, workload, or staffing.

Fairness-constrained bandit algorithms typically study online allocation with fairness concerns, often in non-contextual or combinatorial settings (Claure et al., 2020; Li et al., 2019; Huang et al., 2020), with some contextual extensions (Chen et al., 2020). Unlike Chen et al. (2020), who enforce a shared minimum-rate constraint across arms in expectation, we study agent-specific capacity targets enforced over realized, sequential assignments via a queue. There is also a large literature in general on **regret bounds for fair contextual bandits** (Li et al., 2010; Badanidiyuru et al., 2018; Balseiro et al., 2020; Neu et al., 2022; Blanchet et al., 2024), but this work focuses on algorithmic regret bounds rather than the characterization of assignment policies that we present here.

Across these lines of work, expertise is learned offline, capacity constraints are absent, or the constraints are structurally different from ours (a shared minimum-rate floor or a global cost budget). To our knowledge, no prior work studies the joint problem of online learning of context-dependent expertise under agent-specific, long-run capacity constraints.

6 Discussion and Conclusion

Regarding the **limitations** of our work, we evaluate our framework using proxy agents in controlled settings with relatively low-dimensional contexts. This controlled setup makes it easier to compare contextual and non-contextual assignment policies, but it may understate the complexity of real-world task allocation, where task contexts can be high-dimensional, multimodal, or only partially observed. We also focus on classification tasks with binary reward and immediate feedback. Practical applications may involve delayed outcomes, multi-class classification with asymmetric utilities for different misclassifications, or continuous rewards, and noisier measures of agent expertise. **Future work** can extend the framework in several directions. A natural next step is to study high-dimensional and unstructured contexts, such as text or interaction histories, where the system can learn useful task features before identifying which agents are best suited for each task. Another direction is to move beyond assigning each task to a single agent and allow multiple humans or AI agents to collaborate on the same task, requiring mechanisms for aggregation, disagreement, and capacity-aware coordination. Also of interest is evaluating these policies in real human-AI workflows, where expertise, availability, and behavioral responses to assignment policies may evolve over time.

In summary, we study sequential task allocation under capacity constraints when agents have unknown, context-dependent expertise. We show theoretically and empirically that contextual assignment improves performance by learning which agents are best suited for which tasks, while capacity constraints induce implicit exploration by ensuring that all agents continue to receive assignments. Our results highlight the importance of allocation policies that learn heterogeneous expertise while respecting real-world capacity limits in human-AI collaboration.

AI USE STATEMENT

(This section is **required** and does not count toward the page limit.)

In this work, we used generative AI tools for research execution, specifically to assist with writing and debugging code for data processing and experiments. We have not used generative AI tools for drafting sections of the paper, generating synthetic datasets, or proving mathematical claims, and the remaining required-disclosure tasks are not applicable to this work. Additionally, we used generative AI tools to polish writing and check grammar, and to assist with literature retrieval and discovery, including identifying candidate related work. We have reviewed all AI-assisted work: all AI-generated or AI-modified code was tested and verified for correctness by the authors, and all AI-identified related papers were manually read by the authors to confirm relevance and accuracy of characterization before citation. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

(This section is **recommended** and does not count toward the page limit.)

We do not identify any ethical concerns.

REPRODUCIBILITY STATEMENT

(This section is **recommended** and does not count toward the page limit.)

All datasets used in our experiments are publicly available; dataset details and processing steps are provided in Appendix H.1. Complete proofs and assumptions for our theoretical results are given in Appendices A–G, and implementation details are given in Appendix H. We will release the code and experiment scripts in the final version of the paper.

REFERENCES

Jean Vieira Alves, Diogo Leitão, Sérgio Jesus, Marco OP Sampaio, Javier Liébana, Pedro Saleiro, Mario AT Figueiredo, and Pedro Bizarro. Cost-sensitive learning to defer to multiple experts with workload constraints. *Transactions on Machine Learning Research*, 2024.

Majdi M Alzoubi, Khalid Al-Mugheed, Islam Oweidat, Tahani Alrahbeni, Mohammad M Alnaeem, Amany Anwar Saeed Alabdullah, Sally Mohammed Farghaly Abdelaliem, and Abdelaziz Hendy. Moderating role of relationships between workloads, job burnout, turnover intention, and healthcare quality among nurses. *BMC psychology*, 12(1):495, 2024.

Ashwinkumar Badanidiyuru, Robert Kleinberg, and Aleksandrs Slivkins. Bandits with knapsacks. *Journal of the ACM (JACM)*, 65(3):1–55, 2018.

Santiago Balseiro, Haihao Lu, and Vahab Mirrokni. Dual mirror descent for online allocation problems. In *International Conference on Machine Learning*, pp. 613–628. PMLR, 2020.

Peter Bandi, Oscar Geessink, Quirine Manson, Marcory Van Dijk, Maschenka Balkenhol, Meyke Hermsen, Babak Ehteshami Bejnordi, Byungjae Lee, Kyunghyun Paeng, Aoxiao Zhong, et al. From detection of individual metastases to classification of lymph node status at the patient level: the camelyon17 challenge. *IEEE transactions on medical imaging*, 38(2):550–560, 2018.

Mohsen Bayati, Nima Hamidi, Ramesh Johari, and Khashayar Khosravi. Unreasonable effectiveness of greedy algorithms in multi-armed bandit with many arms. *Advances in Neural Information Processing Systems*, 33:1713–1723, 2020.

Leonard Berlin. Liability of interpreting too many radiographs. *American Journal of Roentgenology*, 175(1):17–22, 2000.

Jose Blanchet, Renyuan Xu, and Zhengyuan Zhou. Delay-adaptive learning in generalized linear contextual bandits. *Mathematics of Operations Research*, 49(1):326–345, 2024.

Olivier Chapelle and Lihong Li. An empirical evaluation of Thompson sampling. *Advances in Neural Information Processing Systems*, 24, 2011.

William G Chase and Herbert A Simon. Perception in chess. *Cognitive psychology*, 4(1):55–81, 1973.

Lingjiao Chen, Matei Zaharia, and James Zou. FrugalGPT: How to use large language models while reducing cost and improving performance. *arXiv preprint arXiv:2305.05176*, 2023.

Yifang Chen, Alex Cuellar, Haipeng Luo, Jignesh Modi, Heramb Nemlekar, and Stefanos Nikolaidis. Fair contextual multi-armed bandits: Theory and experiments. In *Conference on Uncertainty in Artificial Intelligence*, pp. 181–190. PMLR, 2020.

Houston Claure, Yifang Chen, Jignesh Modi, Malte Jung, and Stefanos Nikolaidis. Multi-armed bandits with fairness constraints for distributing resources to human teammates. In *Proceedings of the 2020 ACM/IEEE International Conference on Human-Robot Interaction*, pp. 299–308, 2020.

Giulia DeSalvo, Clara Mohri, Mehryar Mohri, and Yutao Zhong. Budgeted multiple-expert deferral. *arXiv preprint arXiv:2510.26706*, 2025.

Dean Eckles and Maurits Kaptein. Thompson sampling with the online bootstrap. *arXiv preprint arXiv:1410.4009*, 2014.

K Anders Ericsson, Ralf T Krampe, and Clemens Tesch-Römer. The role of deliberate practice in the acquisition of expert performance. *Psychological review*, 100(3):363, 1993.

Kevin W Eva. What every teacher needs to know about clinical reasoning. *Medical Education*, 39(1): 98–106, 2005.

Giorgio Giacinto and Fabio Roli. Dynamic classifier selection. In *Multiple Classifier Systems*, volume 1857 of *Lecture Notes in Computer Science*, pp. 177–189, Berlin, Heidelberg, 2000. Springer. doi: 10.1007/3-540-45014-9_17.

Cleotilde Gonzalez, Kate Donahue, Daniel G Goldstein, Hoda Heidari, Mohammad S Jalali, Beau Schelble, Aarti Singh, and Anita Williams Woolley. Toward a science of human–AI teaming for decision making: A complementarity framework. *PNAS Nexus*, 5(3):pgag030, 2026.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*, 2020.

Zhiming Huang, Yifan Xu, Bingshan Hu, Qipeng Wang, and Jianping Pan. Thompson sampling for combinatorial semi-bandits with sleeping arms and long-term fairness constraints. *arXiv preprint arXiv:2005.06725*, 2020.

Vijay Keswani, Matthew Lease, and Krishnaram Kenthapadi. Towards unbiased and accurate deferral to multiple experts. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, pp. 154–165, 2021.

Vivian Lai, Samuel Carton, Rajat Bhatnagar, Q Vera Liao, Yunfeng Zhang, and Chenhao Tan. Human-AI collaboration via conditional delegation: A case study of content moderation. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, pp. 1–18, 2022.

Fengjiao Li, Jia Liu, and Bo Ji. Combinatorial sleeping bandits with fairness constraints. *IEEE Transactions on Network Science and Engineering*, 7(3):1799–1813, 2019.

Lihong Li, Wei Chu, John Langford, and Robert E Schapire. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international Conference on World Wide Web*, pp. 661–670, 2010.

Yifei Li, Zihui Gao, and Laks VS Lakshmanan. Wiserouter: LLM routing with workload budget constraint. *arXiv preprint arXiv:2607.23765*, 2026.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, et al. Holistic evaluation of language models. *arXiv preprint arXiv:2211.09110*, 2022.

David Madras, Toni Pitassi, and Richard Zemel. Predict responsibly: improving fairness and accuracy by learning to defer. *Advances in Neural Information Processing Systems*, 31, 2018.

Giovanni Montanari, Marco Scarsini, and Vianney Perchet. Learning when to automate: Queue control in human-AI service systems. *arXiv preprint arXiv:2607.06017*, 2026.

Hussein Mozannar and David Sontag. Consistent estimators for learning to defer to an expert. In *International Conference on Machine Learning*, pp. 7076–7087. PMLR, 2020.

Hussein Mozannar, Hunter Lang, Dennis Wei, Prasanna Sattigeri, Subhro Das, and David Sontag. Who should predict? exact algorithms for learning to defer to humans. In *International Conference on Artificial Intelligence and Statistics*, pp. 10520–10545. PMLR, 2023.

Michael Neely. *Stochastic Network Optimization with Application to Communication and Queueing Systems*. Morgan & Claypool Publishers, 2010.

Gergely Neu, Iuliia Olkhovskaia, Matteo Papini, and Ludovic Schwartz. Lifting the information ratio: An information-theoretic analysis of Thompson sampling for contextual bandits. *Advances in Neural Information Processing Systems*, 35:9486–9498, 2022.

Cuong C Nguyen, Thanh-Toan Do, and Gustavo Carneiro. Probabilistic learning to defer: Handling missing expert annotations and controlling workload distribution. In *The Thirteenth International Conference on Learning Representations*, 2025.

Geoff Norman, Meredith Young, and Lee Brooks. Non-analytical models of clinical reasoning: the role of experience. *Medical Education*, 41(12):1140–1145, 2007.

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. *arXiv preprint arXiv:2406.18665*, 2024.

Pranoy Panda, Raghav Magazine, Chaitanya Devaguptapu, Sho Takemori, and Vishal Sharma. Adaptive LLM routing under budget constraints. *arXiv preprint arXiv:2508.21141*, 2025.

Mirabel Reid, Tom Sühr, Claire Vernade, and Samira Samadi. Online decision deferral under budget constraints. *arXiv preprint arXiv:2409.20489*, 2024.

Vildan Salikutluk, Janik Schöpper, Franziska Herbert, Katrin Scheuermann, Eric Frodl, Dirk Balfanz, Frank Jäkel, and Dorothea Koert. An evaluation of situational autonomy for human-AI collaboration in a shared workspace setting. In *Proceedings of the 2024 CHI Conference on human factors in computing systems*, pp. 1–17, 2024.

Ben Shneiderman. Human-centered artificial intelligence: Reliable, safe & trustworthy. *International Journal of Human–Computer Interaction*, 36(6):495–504, 2020.

Wei Song, Zhenya Huang, Cheng Cheng, Weibo Gao, Bihan Xu, GuanHao Zhao, Fei Wang, and Runze Wu. IRT-router: Effective and interpretable multi-LLM routing via item response theory. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15629–15644, 2025.

Kv Aditya Srivatsa, Kaushal Maurya, and Ekaterina Kochmar. Harnessing the power of multiple minds: Lessons learned from llm routing. In *Proceedings of the Fifth Workshop on Insights from Negative Results in NLP*, pp. 124–134, 2024.

Mark Steyvers, Heliodoro Tejeda, Gavin Kerrigan, and Padhraic Smyth. Bayesian modeling of human–AI complementarity. *Proceedings of the National Academy of Sciences*, 119(11):e2111547119, 2022.

Joshua Strong, Emma Sun, Harry Rogers, Helen Higham, and J Alison. Learning to defer: A survey. 2025.

Annette Taberner-Miller. Paretobandit: Budget-paced adaptive routing for non-stationary LLM serving. *arXiv preprint arXiv:2604.00136*, 2026.

Tong Wang, Cynthia Rudin, Finale Doshi-Velez, Yimin Liu, Erica Klampfl, and Perry MacNeille. A Bayesian framework for learning rule sets for interpretable classification. *Journal of Machine Learning Research*, 18(70):1–37, 2017.

I-Cheng Yeh and Che-hui Lien. The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients. *Expert Systems with Applications*, 36(2): 2473–2480, 2009.

Qiaoning Zhang, Matthew L Lee, and Scott Carter. You complete me: Human-AI teams and complementary expertise. In *Proceedings of the 2022 CHI conference on human factors in computing systems*, pp. 1–28, 2022.

Zheng Zhang, Cuong C Nguyen, David Rosewarne, Kevin Wells, and Gustavo Carneiro. Fatigue-aware learning to defer via constrained optimisation. *arXiv preprint arXiv:2604.00904*, 2026.

Nikolas Zöller, Julian Berger, Irving Lin, Nathan Fu, Jayanth Komarneni, Gioele Barabucci, Kyle Laskowski, Victor Shia, Benjamin Harack, Eugene A Chu, et al. Human–AI collectives most accurately diagnose clinical vignettes. *Proceedings of the National Academy of Sciences*, 122(24): e2426153122, 2025.

A Optimal Policy Without Capacity Constraint

Here, we characterize the optimal policy for $A \geq 2$ agents and compare it with non-contextual assignment.

The objective is

$$\max_{\phi} \ \mu^{\phi} = \mathbb{E}_x \left[\sum_{a=1}^{A} P_\phi(a \mid x) \mu_a(x) \right],$$

where $P_\phi(a \mid x)$ denotes the probability that policy ϕ assigns context x to agent a, with $\sum_a P_\phi(a \mid x) = 1$ for all x.

Since the objective is an expectation over x, the assignment decision at one context does not affect assignments at other contexts. Thus, for each fixed x, the policy solves

$$\max_{P_\phi(\cdot|x)} \sum_{a=1}^{A} P_\phi(a \mid x) \mu_a(x), \quad \text{s.t.} \quad \sum_a P_\phi(a \mid x) = 1.$$

Because this objective is linear in $P_\phi(\cdot \mid x)$, an optimum is attained by assigning all probability to an agent with the largest conditional expected reward. Therefore, the optimal policy can be written as

$$\phi^\star(x) = \arg \max_{a \in \{1,\ldots,A\}} \mu_a(x).$$

The resulting marginal reward is

$$\mu^{\phi^\star} = \mathbb{E}_x \left[\max_{a \in \{1,\ldots,A\}} \mu_a(x) \right].$$

We now compare this policy with a non-contextual random assignment policy that assigns each task to agent a with fixed probability P_a, independent of x, where $\sum_a P_a = 1$. Its marginal reward is

$$\mu^{\text{rand}} = \sum_{a=1}^{A} P_a \mu_a = \mathbb{E}_x \left[\sum_{a=1}^{A} P_a \mu_a(x) \right].$$

For every context x, $\max_a \mu_a(x) \geq \sum_{a=1}^{A} P_a \mu_a(x)$, since the maximum is at least any weighted average. Taking expectations over x gives

$$\mu^{\phi^\star} \geq \mu^{\text{rand}}.$$

The inequality is strict when the best agent varies across contexts, and the non-contextual policy sometimes assigns tasks to lower-performing agents. Intuitively, contextual assignment improves by routing each context to the agent with the highest expected reward.

A best fixed-agent policy is a special case of non-contextual assignment. It chooses

$$a^{\max} = \arg \max_{a \in \{1,\dots,A\}} \mu_a$$

and assigns every task to $a^{\max}$, achieving reward $\mu^{\max} = \max_a \mu_a$.

Since $\max_a \mu_a(x) \geq \mu_{a^{\max}}(x)$ for every x, taking expectations gives

$$\mu^{\phi^\star} = \mathbb{E}_x[\max_a \mu_a(x)] \geq \mathbb{E}_x[\mu_{a^{\max}}(x)] = \mu^{\max}.$$

Thus, the optimal contextual policy weakly dominates the best fixed-agent policy, with strict improvement whenever some other agent outperforms $a^{\max}$ on a positive-measure set of contexts.

B Analysis of the Two-Agent Case Without Capacity Constraint

B.1 Optimal Policy

Here, we provide a formal characterization of the two-agent case and quantify the performance gap between contextual and non-contextual policies. Consider two agents with conditional expected rewards $\mu_1(x)$ and $\mu_2(x)$, and define the contextual reward gap

$$\Delta(x) := \mu_1(x) - \mu_2(x).$$

The optimal (unconstrained) policy assigns each context to the agent with the higher conditional expected reward,

$$\phi^\star(x) = \arg \max_{a \in \{1,2\}} \mu_a(x),$$

which is equivalently characterized by selecting agent 1 when $\Delta(x) \geq 0$ and agent 2 otherwise. This induces decision regions

$$\mathcal{R}_1^\star := \{x : \Delta(x) \geq 0\}, \qquad \mathcal{R}_2^\star := \{x : \Delta(x) < 0\}.$$

The marginal reward of the optimal policy is

$$\mu^{\phi^\star} = \int_{x \in \mathcal{R}_1^\star} \mu_1(x)\, P_X(x)\, dx + \int_{x \in \mathcal{R}_2^\star} \mu_2(x)\, P_X(x)\, dx.$$

B.2 Optimal Policy Weakly Dominates Non-contextual Policies

Here we compare the optimal contextual policy with the best context-independent policy. Consider the policy that ignores x and always assigns tasks to the agent with the highest marginal reward $a^{\max} = \arg\max_{a \in \{1,2\}} \mu_a$. Without loss of generality, suppose $a^{\max} = 1$. Then this policy achieves

$$\mu^{\max} = \int \mu_1(x)\, dP_X(x) = \mu_1.$$

The optimal contextual policy weakly dominates this baseline. Its performance gain is

$$\mu^{\phi^\star} - \mu^{\max} = \int_{x \in \mathcal{R}_2^\star} \big(\mu_2(x) - \mu_1(x)\big)\, P_X(x)\, dx = \int_{x \in \mathcal{R}_2^\star} -\Delta(x)\, P_X(x)\, dx \geq 0.$$

The inequality holds because on $\mathcal{R}_2^\star = \{x : \Delta(x) < 0\}$ (i.e., agent 2 outperforms agent 1). Thus, contextual assignment improves over the best fixed-agent policy precisely by switching to agent 2 on these contexts.

B.3 Outperforming Individual Agents

Here, we show that when two agents have complementary expertise across contexts, the combined contextual policy can outperform either individual agent, i.e.,

$$P_X(\Delta(x) > 0) > 0 \quad \text{and} \quad P_X(\Delta(x) < 0) > 0.$$

Then the optimal contextual policy strictly outperforms both fixed-agent policies.

From Appendix B.2, if $\mu_1 \geq \mu_2$, the gain over always assigning to agent 1 is

$$\mu^{\phi^\star} - \mu_1 = \int_{\Delta(x)<0} \big(\mu_2(x) - \mu_1(x)\big)\, dP_X(x) > 0,$$

since $\mu_2(x) > \mu_1(x)$ on $\{\Delta(x) < 0\}$ and this region has positive probability. Similarly, the gain over always assigning to agent 2 is

$$\mu^{\phi^\star} - \mu_2 = \int_{\Delta(x)\geq 0} \big(\mu_1(x) - \mu_2(x)\big)\, dP_X(x) > 0,$$

since $\mu_1(x) > \mu_2(x)$ on $\{\Delta(x) > 0\}$ and this region has positive probability. Therefore,

$$\mu^{\phi^\star} > \max\{\mu_1, \mu_2\}.$$

Thus, when each agent has a positive-measure region of relative advantage, the contextual policy can outperform any single agent by routing each context to the better-performing agent. This is conceptually related to dynamic classifier selection, where classifier complementarity across regions of the feature space enables instance-wise selection to improve over individual classifiers (Giacinto & Roli, 2000).

B.4 Role of Disagreement and Magnitude of Improvement

We now express the improvement from contextual assignment in terms of the disagreement regions. Define

$$\mathcal{D}_1 := \{x : \Delta(x) > 0\}, \qquad \mathcal{D}_2 := \{x : \Delta(x) < 0\},$$

with

$$p_1 := P_X(\mathcal{D}_1), \qquad p_2 := P_X(\mathcal{D}_2).$$

On $\mathcal{D}_1$, agent 1 outperforms agent 2, while on $\mathcal{D}_2$, agent 2 outperforms agent 1. The contextual policy exploits all such disagreement by assigning $\mathcal{D}_1$ to agent 1 and $\mathcal{D}_2$ to agent 2.

Relative to always assigning to agent 1, the improvement is

$$\mu^{\phi^\star} - \mu_1 = \int_{\mathcal{D}_2} \big(\mu_2(x) - \mu_1(x)\big)\, dP_X(x) = \int_{\mathcal{D}_2} -\Delta(x)\, dP_X(x).$$

Similarly, relative to always assigning to agent 2, the improvement is

$$\mu^{\phi^\star} - \mu_2 = \int_{\mathcal{D}_1} \big(\mu_1(x) - \mu_2(x)\big)\, dP_X(x) = \int_{\mathcal{D}_1} \Delta(x)\, dP_X(x).$$

Equivalently, these improvements can be written as

$$\mu^{\phi^\star} - \mu_1 = p_2 \cdot \mathbb{E}\big[-\Delta(X) \mid X \in \mathcal{D}_2\big],$$

and

$$\mu^{\phi^\star} - \mu_2 = p_1 \cdot \mathbb{E}\big[\Delta(X) \mid X \in \mathcal{D}_1\big].$$

Thus, relative to a fixed-agent assignment, the improvement from contextual assignment depends both on the probability mass of contexts in which the fixed agent is worse than the alternative agent and on the magnitude of the contextual reward gap within those contexts.

C Optimal Policy under Capacity Constraints

Here, we show the optimal policy for $A \geq 2$ agents under capacity constraints and establish its performance relative to non-contextual assignment.

Our goal is

$$\max_{\phi} \mathbb{E}_x \left[\sum_{a=1}^{A} P_\phi(a \mid x) \mu_a(x) \right] \quad \text{s.t.} \quad \mathbb{E}_x[P_\phi(a \mid x)] = \alpha_a, \ \forall a,$$

where $\sum_{a=1}^{A} \alpha_a = 1$ and $P_\phi(a \mid x)$ denotes the probability that policy ϕ assigns context x to agent a, with $\sum_a P_\phi(a \mid x) = 1$ for all x.

Introduce Lagrange multipliers $\lambda_1, \ldots, \lambda_A$ and consider the Lagrangian

$$\begin{aligned}
\mathcal{L}(\phi, \lambda) &= \mathbb{E}_x \left[\sum_{a=1}^{A} P_\phi(a \mid x) \mu_a(x) \right] - \sum_{a=1}^{A} \lambda_a \left(\mathbb{E}_x[P_\phi(a \mid x)] - \alpha_a \right) \\
&= \mathbb{E}_x \left[\sum_{a=1}^{A} P_\phi(a \mid x) \mu_a(x) \right] - \sum_{a=1}^{A} \lambda_a \mathbb{E}_x[P_\phi(a \mid x)] + \sum_{a=1}^{A} \lambda_a \alpha_a \\
&= \mathbb{E}_x \left[\sum_{a=1}^{A} P_\phi(a \mid x) \big(\mu_a(x) - \lambda_a \big) \right] + \sum_{a=1}^{A} \lambda_a \alpha_a.
\end{aligned}$$

For fixed λ, the second term is constant in ϕ. Since the objective is an expectation over x, the assignment decision at one context does not affect others. Thus, we can determine the optimal assignment separately for each x:

$$\max_{P_\phi(\cdot|x)} \sum_{a=1}^{A} P_\phi(a \mid x) \big(\mu_a(x) - \lambda_a \big), \quad \text{s.t.} \ \sum_a P_\phi(a \mid x) = 1,$$

since this objective is linear in $P_\phi(\cdot \mid x)$, an optimum is attained by assigning all probability to an agent with the largest adjusted reward; that is, $P_\phi(a \mid x) \in \{0, 1\}$ for all a. Therefore, the optimal policy can be written as a deterministic rule:

$$\phi^\star(x) = \arg \max_{a \in \{1, \ldots, A\}} \big\{ \mu_a(x) - \lambda_a \big\}.$$

There exist constants $\{\lambda_a\}$ such that the resulting policy satisfies the constraints

$$\mathbb{E}_x[P_{\phi^\star}(a \mid x)] = \alpha_a.$$

Under the continuous-context assumption stated in Section 2.3, this policy can be chosen deterministically while satisfying the capacity constraints exactly.

To compare with the non-contextual assignment, define $g_a(x) := \mu_a(x) - \lambda_a$. Then the optimal policy is

$$\mu^{\phi^\star} = \mathbb{E}_x \Big[\max_a g_a(x) \Big] + \sum_{a=1}^{A} \lambda_a \alpha_a,$$

while the non-contextual policy yields

$$\begin{aligned}\mu^{\text{rand}} &= \sum_{a=1}^{A} \alpha_a \mu_a \\ &= \sum_{a=1}^{A} \alpha_a \mathbb{E}_x[\mu_a(x)] \\ &= \mathbb{E}_x\left[\sum_{a=1}^{A} \alpha_a \mu_a(x)\right] \\ &= \mathbb{E}_x\left[\sum_{a=1}^{A} \alpha_a \big(g_a(x) + \lambda_a\big)\right] \\ &= \mathbb{E}_x\left[\sum_{a=1}^{A} \alpha_a g_a(x)\right] + \sum_{a=1}^{A} \lambda_a \alpha_a.\end{aligned}$$

Since for any numbers $\{z_a\}$ and weights $\{\alpha_a\}$ with $\sum_a \alpha_a = 1$,

$$\max_a z_a \geq \sum_{a=1}^{A} \alpha_a z_a,$$

we have pointwise

$$\max_a g_a(x) \geq \sum_{a=1}^{A} \alpha_a g_a(x).$$

Taking expectations yields $\mu^{\phi^\star} \geq \mu^{\text{rand}}$.

The inequality is strict whenever the best agent varies across contexts, so that the non-contextual policy averages over suboptimal agents while $\phi^\star$ selects the best agent at each context.

Finally, the constants λ_a can be interpreted as shadow prices associated with the capacity constraints. They quantify the marginal value of relaxing each constraint and characterize how the optimal policy trades off reward and capacity across agents.

D Two-Agent Case under Capacity Constraints

Throughout this section, we distinguish two cases. We refer to the *non-dominant case* as the setting where two agents exhibit heterogeneous expertise across contexts, i.e., $P_X(\Delta(x) > 0) > 0$ and $P_X(\Delta(x) < 0) > 0$. We refer to the *dominant case* as the setting where one agent weakly dominates the other, i.e., $\mu_1(x) \geq \mu_2(x)$ for all x.

D.1 Optimal Policy under Capacity Constraints

Here we show that the optimal constrained policy assigns agent 1 to the α-mass of contexts with the largest values of $\Delta(x)$, where $\Delta(x) = \mu_1(x) - \mu_2(x)$.

Let the target workload capacities be $\alpha_1 = \alpha$ and $\alpha_2 = 1 - \alpha$. Any policy ϕ induces regions

$$\mathcal{R}_1 := \{x \in \mathcal{X} : \phi(x) = 1\}, \qquad \mathcal{R}_2 := \mathcal{X} \setminus \mathcal{R}_1,$$

with

$$\int_{x \in \mathcal{R}_1} P_X(x)\, dx = \alpha.$$

Any feasible policy induces a region $\mathcal{R}_1$ with $P_X(\mathcal{R}_1) = \alpha$, assigning agent 1 to $\mathcal{R}_1$ and agent 2 to its complement.

The expected reward is

$$\begin{aligned}\mu^{\phi} &= \int_{x\in\mathcal{R}_1} \mu_1(x)P_X(x)\,dx + \int_{x\in\mathcal{R}_2} \mu_2(x)P_X(x)\,dx \\ &= \mu_2 + \int_{x\in\mathcal{R}_1} \Delta(x)P_X(x)\,dx,\end{aligned}$$

Thus, the problem reduces to selecting a set $\mathcal{R}_1$ of mass α that maximizes

$$\int_{\mathcal{R}_1} \Delta(x)\,dP_X(x).$$

This is solved by choosing the α-mass of contexts with the largest values of $\Delta(x)$. Therefore, there exists a threshold τ_α such that

$$\mathcal{R}_1^\star = \{x : \Delta(x) \geq \tau_\alpha\}, \qquad \mathcal{R}_2^\star = \mathcal{X} \setminus \mathcal{R}_1^\star.$$

This threshold characterization is consistent with the general Lagrangian solution in Appendix C. There, the optimal policy takes the form

$$\phi^\star(x) = \arg\max_{a\in\{1,2\}} \{\mu_a(x) - \lambda_a\}.$$

For two agents, this assigns to agent 1 whenever

$$\mu_1(x) - \lambda_1 \geq \mu_2(x) - \lambda_2,$$

which can be rewritten as

$$\Delta(x) := \mu_1(x) - \mu_2(x) \geq \lambda_1 - \lambda_2.$$

Thus, the threshold τ_α corresponds to the shadow-price difference $\lambda_1 - \lambda_2$, determined by the capacity constraint.

The optimal policy therefore assigns agent 1 to the top α-mass of contexts ranked by $\Delta(x)$, and agent 2 to the remainder. The resulting marginal reward is

$$\mu^{\phi^\star} = \mu_2 + \int_{x\in\mathcal{R}_1^\star} \Delta(x)P_X(x)\,dx.$$

D.2 Constrained Optimal Policy Weakly Dominates Non-contextual Policy

Here we establish that this policy weakly dominates the random policy in both the *non-dominant case* and the *dominant case*. As shown in Appendix D.1, the constrained optimal policy has reward

$$\mu^{\phi^\star} = \mu_2 + \int_{x\in\mathcal{R}_1^\star} \Delta(x)P_X(x)\,dx.$$

By contrast, the non-contextual (i.e., random) policy has reward

$$\mu^{\text{rand}} = \alpha\mu_1 + (1-\alpha)\mu_2 = \mu_2 + \alpha\int \Delta(x)P_X(x)\,dx.$$

Then, the degree to which the optimal policy improves over the random policy is

$$\begin{aligned}\mu^{\phi^\star} - \mu^{\text{rand}} &= \int_{x\in\mathcal{R}_1^\star} \Delta(x)P_X(x)\,dx - \alpha\int \Delta(x)P_X(x)\,dx \\ &= (1-\alpha)\int_{x\in\mathcal{R}_1^\star} \Delta(x)P_X(x)\,dx - \alpha\int_{x\in\mathcal{R}_2^\star} \Delta(x)P_X(x)\,dx \;\geq\; 0.\end{aligned}$$

D.3 Outperforming Individual Agents

Here, we show that, for the *non-dominant case*, the combined policy can outperform either individual agent when performance varies across contexts.

Suppose

$$P_X(\Delta(x) > 0) > 0 \quad \text{and} \quad P_X(\Delta(x) < 0) > 0.$$

Let $\alpha = P_X(\Delta(x) \geq 0)$. Then the optimal set becomes

$$\mathcal{R}_1^\star = \{x : \Delta(x) \geq 0\},$$

so each context is assigned to the better-performing agent.

The resulting policy assigns each context x to the better-performing agent. Its reward becomes:

$$\begin{aligned}
\mu^{\phi^\star} &= \int_{\Delta(x)\geq 0} \mu_1(x)\, dP_X(x) + \int_{\Delta(x)<0} \mu_2(x)\, dP_X(x) \\
&= \mu_1 + \int_{\Delta(x)<0} \big(\mu_2(x) - \mu_1(x)\big)\, dP_X(x) \\
&> \mu_1,
\end{aligned}$$

since $\mu_2(x) > \mu_1(x)$ on $\{\Delta(x) < 0\}$.

Similarly:

$$\mu^{\phi^\star} = \mu_2 + \int_{\Delta(x)\geq 0} \big(\mu_1(x) - \mu_2(x)\big) P_X(x)\, dx > \mu_2.$$

so

$$\mu^{\phi^\star} > \max\{\mu_1, \mu_2\}.$$

D.4 Role of Disagreement and Magnitude of Improvement

Define the disagreement region between two agents

$$\mathcal{D} := \{x \in \mathcal{X} : \Delta(x) \neq 0\},$$

and partition it into

$$\mathcal{D}_1 := \{x : \Delta(x) > 0\}, \qquad \mathcal{D}_2 := \{x : \Delta(x) < 0\}.$$

Let

$$p_1 := P_X(\mathcal{D}_1), \quad p_2 := P_X(\mathcal{D}_2), \quad t := P_X(\mathcal{D}) = p_1 + p_2.$$

On $\mathcal{D}_1$, agent 1 strictly outperforms agent 2, while the reverse holds on $\mathcal{D}_2$. The optimal policy exploits this heterogeneity by allocating capacity to contexts with the largest values of $\Delta(x)$, whereas the non-contextual random policy ignores this structure. As shown in Appendix D.2, the improvement over the non-contextual policy is

$$\mu^{\phi^\star} - \mu^{\text{rand}} = \int_{\mathcal{R}_1^\star} \Delta(x)\, dP_X(x) - \alpha \int \Delta(x)\, dP_X(x) \;\geq\; 0.$$

When capacity allows full exploitation of disagreement, i.e., $p_1 \leq \alpha \leq 1 - p_2$, the policy assigns $\mathcal{D}_1$ to agent 1 and $\mathcal{D}_2$ to agent 2.

In this case, the improvement over the non-contextual random policy can be written as

$$\mu^{\phi^\star} - \mu^{\text{rand}} = (1-\alpha) \int_{\mathcal{D}_1} \Delta(x)\, dP_X(x) + \alpha \int_{\mathcal{D}_2} -\Delta(x)\, dP_X(x).$$

Equivalently,

$$\mu^{\phi^\star} - \mu^{\text{rand}} = (1-\alpha)p_1 \cdot \mathbb{E}\big[\Delta(X) \mid X \in \mathcal{D}_1\big] + \alpha p_2 \cdot \mathbb{E}\big[-\Delta(X) \mid X \in \mathcal{D}_2\big].$$

Thus, the magnitude of improvement is determined by both the size of the disagreement regions and the average contextual reward gap within those regions, weighted by the capacity α.

D.5 Equivalence of Learned and Random Non-Contextual Policies

Consider any non-contextual policy whose assignment decision does not depend on the current context x_t. This includes policies that learn only marginal agent performance, for example, policies that maintain estimates $\hat{\mu}_{a,t}$ of μ_a and choose

$$a_t = \arg\max_a \{\hat{\mu}_{a,t} - \eta Q_{t,a}\}.$$

Let H_t denote the history before the reward at round t. Because the policy is non-contextual, conditioning on $a_t = a$ does not change the distribution of x_t. Therefore,

$$\mathbb{E}[r_{t,a} \mid H_t, a_t = a] = \mathbb{E}_{x_t}[\mathbb{E}[r_{t,a} \mid x_t]] = \mathbb{E}_{x_t}[\mu_a(x_t)] = \mu_a.$$

So we have,

$$\mathbb{E}[r_{t,a_t} \mid H_t] = \sum_a \Pr(a_t = a \mid H_t)\mu_a.$$

Averaging over time gives

$$\frac{1}{T}\sum_{t=1}^{T} \mathbb{E}[r_{t,a_t}] = \sum_a \left(\frac{1}{T}\sum_{t=1}^{T} \Pr(a_t = a)\right)\mu_a.$$

As the capacity mechanism enforces the long-run assignment capacities

$$\frac{1}{T}\sum_{t=1}^{T} \Pr(a_t = a) \to \alpha_a,$$

then the limiting expected reward of the learned non-contextual policy is

$$\sum_a \alpha_a \mu_a.$$

The random non-contextual baseline assigns agent a with probability α_a, independently of x_t, and therefore has the same expected reward:

$$\mathbb{E}[r_{t,a_t}] = \sum_a \alpha_a \mathbb{E}_{x_t}[\mu_a(x_t)] = \sum_a \alpha_a \mu_a.$$

Thus, under the same long-run assignment capacities, learned and random non-contextual policies are asymptotically equivalent in expected reward. Finite-sample realizations may differ due to initialization, queue transients, tie-breaking, and reward noise, but without conditioning on x_t, learned marginal accuracies cannot improve performance through context-dependent sorting.

E Contextual Reward Model Implementation Details

This section provides implementation details for the contextual reward model updates used in the online assignment procedure described in Section 3.2.

Algorithm 1 Online capacity-constrained assignment

1: **Input:** Capacities $\{\alpha_a\}$, queue penalty η
2: Initialize estimates $\mu_{a,1}(\cdot)$ and queues $Q_{1,a} = 0$ for all a
3: **for** $t = 1, 2, \ldots$ **do**
4: Observe context x_t
5: Select agent $a_t = \arg\max_a \{\mu_{a,t}(x_t) - \eta Q_{t,a}\}$
6: Observe reward r_{t,a_t} and update context models and posterior distributions
7: Update queues $Q_{t+1,a} = \left[Q_{t,a} + \mathbb{1}\{a_t = a\} - \alpha_a\right]_+$
8: **end for**

E.1 Logistic Contextual Model: Greedy and Thompson Sampling

Here we describe the posterior approximation and online update procedure used for Thompson sampling in logistic models. For each agent a, rewards are modeled as

$$r_{t,a} \sim \text{Bernoulli}(\sigma(\theta_a^\top x_t)),$$

where $\sigma(z) = (1 + e^{-z})^{-1}$.

For each agent a, let $\mathcal{D}_{a,t} = \{(x_s, r_{j,a}) : s \le t,\ a_j = a\}$ denote the bandit feedback observed for that agent. Given $\mathcal{D}_{a,t}$, the posterior over θ_a is intractable. Following Chapelle & Li (2011), we use a Laplace approximation centered at the maximum a posteriori (MAP) estimate. The posterior is approximated as

$$\theta_a \mid \mathcal{D}_{a,t} \approx \mathcal{N}(\hat{\theta}_{a,t}, \Sigma_{a,t}),$$

where $\hat{\theta}_{a,t}$ is the MAP estimate and $\Sigma_{a,t}$ is the inverse Hessian of the negative log-posterior.

To enable efficient online updates, we maintain $\hat{\theta}_{a,t}$ and $\Sigma_{a,t}$ incrementally. For a new observation $(x_t, r_{t,a})$, define

$$\hat{p}_t = \sigma(\hat{\theta}_{a,t}^\top x_t), \quad w_t = \hat{p}_t(1 - \hat{p}_t).$$

The covariance is updated using a Sherman–Morrison step:

$$\Sigma_{a,t+1} = \Sigma_{a,t} - \frac{w_t \Sigma_{a,t} x_t x_t^\top \Sigma_{a,t}}{1 + w_t x_t^\top \Sigma_{a,t} x_t},$$

and the mean is updated as

$$\hat{\theta}_{a,t+1} = \hat{\theta}_{a,t} + \Sigma_{a,t+1}(r_{t,a} - \hat{p}_t) x_t.$$

At each round, Thompson sampling draws

$$\tilde{\theta}_{a,t} \sim \mathcal{N}(\hat{\theta}_{a,t}, \kappa^2 \Sigma_{a,t}),$$

where $\kappa > 0$ controls exploration, and finally uses

$$\mu_{a,t}(x_t) = \sigma(\tilde{\theta}_{a,t}^\top x_t)$$

for decision making.

In our implementation, we initialize the posterior mean at zero, $\hat{\theta}_{a,1} = 0$, and the covariance as $\Sigma_{a,1} = \gamma_{\text{prior}}^{-1} I$. We set $\gamma_{\text{prior}} = 1$ and use exploration scale $\kappa = 0.5$ for Thompson sampling. For numerical stability, the covariance matrix is symmetrized after each update, and w_t is clipped to the interval $[10^{-4}, 0.25]$.

For the greedy logistic policy, we use the same online MAP update for $\hat{\theta}_{a,t}$, but do not sample from the approximate posterior. Instead, the policy uses the plug-in estimate $\mu_{a,t}(x_t) = \sigma(\hat{\theta}_{a,t}^\top x_t)$ for decision making.

E.2 Tree-Based Contextual Model: Greedy and Bootstrap Thompson Sampling

Here we provide the implementation details of the tree-based contextual reward model used in our online policies. For each agent a, we maintain an ensemble of regression trees that predicts the realized reward $r_{t,a}$ from the context x_t. The model is updated online using only the observations assigned to that agent.

For each agent, we use an ensemble of $B = 20$ decision-tree regressors, each with maximum depth 3 and minimum leaf size 10. The model stores all observed pairs $(x_t, r_{t,a})$ for that agent. To reduce computational cost, the ensemble is refit every 20 observed updates rather than after every round. At each refit, each tree is trained on a bootstrap sample of the agent's observed data.

For the greedy tree policy, the estimated reward is the average prediction across trees: $\mu_{a,t}(x_t) = \frac{1}{B}\sum_{b=1}^{B} f_{a,t}^{(b)}(x_t)$, where $f_{a,t}^{(b)}$ denotes the b-th tree for agent a at time t. Before the ensemble is first fitted, we use a prior mean of 0.5.

For Thompson sampling, we approximate posterior sampling using bootstrap randomness. At each round t, we sample one tree uniformly at random from the ensemble and use its prediction as the sampled reward estimate: $\tilde{\mu}_{a,t}(x_t) = f_{a,t}^{(b)}(x_t), b \sim \text{Uniform}\{1, \ldots, B\}$.. If the ensemble has not yet been fitted, we sample an initial reward estimate uniformly from $[0, 1]$. All tree predictions are clipped to lie in $[0, 1]$.

F Capacity Constraints and Queue-Based Implementation

Here, we provide a detailed description of the capacity-constrained assignment problem and the queue-based method used to enforce long-run constraints. Let $\alpha_a \in [0, 1]$ denote the target long-run fraction of tasks assigned to agent a, with $\sum_{a=1}^{A} \alpha_a = 1$. For each round t, define the assignment indicator $I_{t,a} := \mathbb{1}\{a_t = a\}$, which equals one if task t is assigned to agent a. Following the long-term time-average constraint framework of Neely (2010), the policy is required to satisfy

$$\limsup_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \mathbb{E}[I_{t,a}] \leq \alpha_a, \qquad a = 1, \ldots, A.$$

When the full workload is allocated each round and $\sum_{a=1}^{A} \alpha_a = 1$, these upper-bound constraints bind at optimum, so the long-run expected fraction of tasks handled by agent a matches α_a.

To enforce these constraints in an online setting, we introduce a virtual queue $Q_{t,a}$ (notation first proposed by (Neely, 2010)) for each agent, which tracks the cumulative deviation between realized assignments and the target capacity. The queue evolves as

$$Q_{t+1,a} = \big[Q_{t,a} + I_{t,a} - \alpha_a\big]_+, \quad a = 1, .., A,$$

where $[\cdot]_+ = \max(\cdot, 0)$. $Q_{t,a}$ increases when agent a is over-assigned relative to its target capacity, and remains small when assignments are balanced.

At each round t, the agent is selected by trading off predicted reward and queue pressure:

$$\phi(x_t) = \arg \max_{a\in\{1,\ldots,A\}} \{\mu_{a,t}(x_t) - \eta Q_{t,a}\},$$

where $\mu_a(x_t)$ is the agent a's expected reward on context x_t and $\eta \geq 0$ controls the trade-off between reward maximization and constraint satisfaction. Agents with larger queues are penalized, which encourages the policy to rebalance assignments over time while still prioritizing high-reward decisions. In this sense, the queue lengths act as dynamic shadow prices for capacity constraints.

G Regret Bounds for Penalized Contextual Thompson Sampling

Throughout, rewards lie in $[0, 1]$, the estimates satisfy $\mu_{a,t}(\cdot) \in [0, 1]$ (logistic outputs and clipped tree outputs), capacities satisfy $\sum_a \alpha_a = 1$, and $Q_{1,a} = 0$ as in Algorithm 1. $I_{t,a} = \mathbb{1}\{a_t = a\}$ and $[\cdot]_+$ are as in equation 1, and the selection rule is equation 2 with a fixed tie-breaking rule. $\mathcal{F}_{t-1}$ denotes the history $(x_s, a_s, r_{s,a_s})_{s<t}$, so $Q_t = (Q_{t,a})_a$ is $\mathcal{F}_{t-1}$-measurable. Let $\phi^\star$ be the optimal constrained policy of Theorem 2.1 and $\mu^\star := \mu^{\phi^\star}$ its value. We separate two quantities:

$$R_T := \mathbb{E}\Big[\sum_{t=1}^{T} \big(\mu^\star - \mu_{a_t}(x_t)\big)\Big],$$

$$\tilde{R}_T := \mathbb{E}\Big[\sum_{t=1}^{T} \rho_t\Big], \qquad \rho_t := \max_a \big\{\mu_a(x_t) - \eta Q_{t,a}\big\} - \big\{\mu_{a_t}(x_t) - \eta Q_{t,a_t}\big\} \;\geq\; 0.$$

R_T is the reward regret against the constrained optimum. $\tilde{R}_T$ is the regret of the selection rule with respect to the penalized reward $\mu_a(x) - \eta Q_{t,a}$, the quantity that standard contextual bandit theory controls. Lemma G.1 and Theorem G.1 hold for fixed reward functions $\{\mu_a\}$; in the Bayesian setting of Proposition G.1 they hold conditionally on the true parameter, and hence also in expectation over the prior.

Lemma G.1 (Deterministic queue bound). *For any $\eta > 0$ and any estimates $\mu_{a,t}(\cdot) \in [0,1]$,*

$$Q_{t,a} \;\le\; C_\eta := A(1+\eta^{-1}) + 2 \qquad \textit{for all } t \ge 1 \textit{ and all } a.$$

Proof. Let $m_t := \min_a Q_{t,a}$ and $M_t := \max_{s \le t} m_s$, with $M_0 := 0$.

Step 1 (a selected queue is close to the minimum). If $a_t = a$, then $\mu_{a,t}(x_t) - \eta Q_{t,a} \ge \mu_{b,t}(x_t) - \eta Q_{t,b}$ for $b \in \arg\min_b Q_{t,b}$, hence $Q_{t,a} \le m_t + \eta^{-1}$. Unselected queues do not increase. By induction from $Q_{1,a} = 0$,

$$\max_a Q_{t,a} \;\le\; M_{t-1} + \eta^{-1} + 1 \qquad \text{for all } t. \tag{3}$$

Step 2 (the minimum is controlled by the maximum). Since $\sum_a \alpha_a = 1$,

$$\sum_a Q_{t+1,a} - \sum_a Q_{t,a} \;=\; \sum_{a \ne a_t} \big[\alpha_a - Q_{t,a}\big]_+,$$

so the total queue length is non-decreasing and increases only at rounds with $m_t < 1$. Fix t with $\sum_a Q_{t,a} > 0$ and let $s < t$ be the last round at which the total increased, so $\sum_a Q_{t,a} = \sum_a Q_{s+1,a}$. An agent b with $Q_{s,b} = m_s < 1$ has $Q_{s+1,b} < 2$, and every other queue is at most $M_s + \eta^{-1} + 1$ by equation 3. Hence

$$A\, m_t \;\le\; \sum_a Q_{t,a} \;\le\; (A-1)\big(M_{t-1} + \eta^{-1} + 1\big) + 2. \tag{4}$$

Step 3 (fixed point). Let $\bar{C}_\eta := (A-1)(1+\eta^{-1}) + 2$. If $M_{t-1} \le \bar{C}_\eta$, then equation 4 gives $m_t \le \bar{C}_\eta$, so $M_t \le \bar{C}_\eta$. Since $M_0 = 0$, $M_t \le \bar{C}_\eta$ for all t, and equation 3 yields $Q_{t,a} \le \bar{C}_\eta + \eta^{-1} + 1 = C_\eta$. □

Theorem G.1 (Finite-time guarantees). *For any $\eta > 0$ and $T \ge 1$:*

(a) (Capacity) *For every agent a, deterministically,*

$$\frac{1}{T}\sum_{t=1}^{T} I_{t,a} - \alpha_a \;\le\; \frac{C_\eta}{T}.$$

Since these deviations sum to zero over a, also $\big|\frac{1}{T}\sum_t I_{t,a} - \alpha_a\big| \le (A-1)C_\eta/T$.

(b) (Reward) $R_T \le \eta T + \tilde{R}_T$.

Proof. (a) Since $[z]_+ \ge z$ and $Q_{1,a} = 0$, $Q_{T+1,a} \ge \sum_{t \le T}(I_{t,a} - \alpha_a)$; apply Lemma G.1.

(b) Let $L_t := \frac{1}{2}\sum_a Q_{t,a}^2$ and $V := \eta^{-1}$.

Drift. Using $[z]_+^2 \le z^2$ and $\frac{1}{2}\sum_a (I_{t,a} - \alpha_a)^2 = \frac{1}{2}\big(1 - 2\alpha_{a_t} + \sum_a \alpha_a^2\big) \le 1$,

$$L_{t+1} - L_t \;\le\; 1 + \sum_a Q_{t,a}(I_{t,a} - \alpha_a) \;=\; 1 + Q_{t,a_t} - \sum_a \alpha_a Q_{t,a}. \tag{5}$$

Comparison with the constrained optimum. Let $a_t^\star \sim P_{\phi^\star}(\cdot \mid x_t)$ be the action of the optimal constrained policy, drawn independently of the algorithm. By definition of ρ_t,

$$\eta Q_{t,a_t} - \mu_{a_t}(x_t) \;\le\; \eta Q_{t,a_t^\star} - \mu_{a_t^\star}(x_t) + \rho_t.$$

Multiplying by V and combining with equation 5,

$$L_{t+1} - L_t - V\mu_{a_t}(x_t) \;\le\; 1 + Q_{t,a_t^\star} - \sum_a \alpha_a Q_{t,a} - V\mu_{a_t^\star}(x_t) + V\rho_t. \tag{6}$$

Conditioning. Q_t is $\mathcal{F}_{t-1}$-measurable and x_t is independent of $\mathcal{F}_{t-1}$, so

$$\mathbb{E}\big[Q_{t,a_t^\star} \,\big|\, \mathcal{F}_{t-1}\big] = \sum_a Q_{t,a}\, \mathbb{E}_x\big[P_{\phi^\star}(a \mid x)\big] = \sum_a \alpha_a Q_{t,a}, \qquad \mathbb{E}\big[\mu_{a_t^\star}(x_t) \,\big|\, \mathcal{F}_{t-1}\big] = \mu^\star.$$

Taking expectations in equation 6, summing over $t \le T$, and using $L_1 = 0 \le L_{T+1}$ gives $VR_T \le T + V\tilde{R}_T$, i.e. $R_T \le \eta T + \tilde{R}_T$. □

Proposition G.1 (Bounding $\tilde{R}_T$ for Thompson sampling)**.** *Let $\theta^\star := (\theta_1^\star, \dots, \theta_A^\star)$ be the true agent parameters of Section 3, drawn from a known prior, and let $\theta_t := (\tilde{\theta}_{1,t}, \dots, \tilde{\theta}_{A,t})$ be the posterior sample used at round t. Suppose $\mu_a(x) = \mu_a(x; \theta^\star)$, $r_{t,a} \in \{0, 1\}$, and $\mu_{a,t}(\cdot) = \mu_a(\cdot; \theta_t)$ with θ_t drawn from the exact posterior given $\mathcal{F}_{t-1}$. Then $\tilde{R}_T$ satisfies the Bayesian regret bounds of Neu et al. (2022): $O(\sqrt{AT\,H(\theta^\star)})$ for a finite-entropy prior, and $\tilde{O}(A\sqrt{dT})$ for the logistic model of Section 3 with Lipschitz logits, where the joint parameter $\theta^\star \in \mathbb{R}^{Ad}$.*

Proof. Let $\tilde{x}_t := (x_t, Q_t)$ and $\bar{\mu}_a(\tilde{x}_t; \theta) := \mu_a(x_t; \theta) - \eta Q_{t,a}$, and let $a_t^\dagger := \arg\max_a \bar{\mu}_a(\tilde{x}_t; \theta^\star)$, so that $\rho_t = \bar{\mu}_{a_t^\dagger}(\tilde{x}_t; \theta^\star) - \bar{\mu}_{a_t}(\tilde{x}_t; \theta^\star)$. Since Q_t is $\mathcal{F}_{t-1}$-measurable, $\tilde{x}_t$ is a context selected by an adaptive adversary, a setting covered by Neu et al. (2022). The penalty is known and independent of $\theta^\star$, so the posterior over $\theta^\star$ coincides with that of the unpenalized problem. Hence $a_t = \arg\max_a \bar{\mu}_a(\tilde{x}_t; \theta_t)$ is exact Thompson sampling for the reward $\bar{\mu}$, and $\tilde{R}_T$ is its Bayesian regret. Because θ_t and $\theta^\star$ have the same distribution given $\mathcal{F}_{t-1}$ and $\tilde{x}_t$, the pairs (a_t, θ_t) and $(a_t^\dagger, \theta^\star)$ are identically distributed, giving the decoupling identity

$$\mathbb{E}\big[\bar{\mu}_{a_t^\dagger}(\tilde{x}_t; \theta^\star) - \bar{\mu}_{a_t}(\tilde{x}_t; \theta^\star)\big] = \mathbb{E}\big[\bar{\mu}_{a_t}(\tilde{x}_t; \theta_t) - \bar{\mu}_{a_t}(\tilde{x}_t; \theta^\star)\big] = \mathbb{E}\big[\mu_{a_t}(x_t; \theta_t) - \mu_{a_t}(x_t; \theta^\star)\big],$$

in which the per-action shift $\eta Q_{t,a}$ cancels, so the information-ratio bound for rewards in $[0, 1]$ applies unchanged. □

Corollary G.1. *Under the assumptions of Proposition G.1 with $\eta = \sqrt{A/T}$,*

$$R_T = O\big(\sqrt{AT}\,(1 + \sqrt{H(\theta^\star)})\big),$$

and every agent's realized assignment fraction is within $O(\sqrt{A/T})$ of α_a (for $T \geq A$).

Remarks.

(i) Lemma G.1 and Theorem G.1(a) require only $\mu_{a,t} \in [0, 1]$, so they hold for all four deployed policies (including greedy and the Laplace/bootstrap approximations) at any fixed η. At $\eta = 0.5$ and $A = 2$, $C_\eta = 8$, giving a deviation of at most $8/T$: 0.004 for $T \geq 2000$ and 0.037 for MMLU ($T = 217$). The observed maximum deviations are 0.0011 and 0.0113 (Table 5). The lemma assumes every agent carries a queue with $\sum_a \alpha_a = 1$ and therefore does not cover the unconstrained-agent extension of Appendix H.6.

(ii) Theorem G.1(b) is sublinear only if $\eta \to 0$ with T. At fixed η, the ηT term reflects the granularity of the queue-implied shadow prices $\eta Q_{t,a}$, which move in increments of order η. Appendix H.8 reports the corresponding reward–capacity trade-off empirically; the deployed $\eta = 0.5$ accepts an $O(\eta)$ per-round reward term in exchange for tighter capacity tracking.

(iii) Proposition G.1 does not cover greedy selection or approximate posteriors. For those, Theorem G.1(b) still holds with $\tilde{R}_T$ left as the estimation-error term.

H Experiment Details and Additional Results

All experiments are run on 2 x AMD Epyc 7313 (16 cores) CPUs. No GPU or major RAM was required.

H.1 Dataset Details

Here we provide additional details on the datasets used in Section 4 and Appendix H.2. The tabular datasets are publicly available from the following sources: Bank Marketing (i.e., **Bank**), Default of Credit Card Clients (i.e., **Credit**), Vehicle Coupon Recommendation (i.e., **Coupon**), and Cardiovascular Disease (i.e., **Cardio**), **MMLU**,**ImageNet16H** , CivilComments-WILDS(i.e., **Civil**)), and **Camelyon17**.

Table 1 summarizes the datasets and agents' marginal performance.

Table 1: Summary of datasets and agent performance. We use a subsample of each dataset for online evaluation under a train–test split; the resulting number of timesteps/tasks is denoted by T and varies across datasets.

Dataset	# Features	# Timesteps/Tasks	Agent 1 Acc.	Agent 2 Acc.
Bank	15	12,357	0.91	0.53
Credit	23	9,000	0.82	0.67
Coupon	24	3,624	0.72	0.57
Cardio	11	21,000	0.74	0.58
MMLU	1	217	0.32	0.47
ImageNet16H	1	2000 (200 images × duplicated 10x)	0.77	0.86
Camelyon17	1	2000	0.69	0.69
Civil	10	5880	0.58	0.56

For the **MMLU** experiments, we use multiple choice questions from 2 categories: *college chemistry* and *US foreign policy*. We use the sentence-transformers/all-MiniLM-L6-v2 model from HuggingFace to transform the text to a 384-dimensional embedding. We then apply principal component analysis to the high-dimensional embedding to identify the first principal component. This is used as the single context feature for the learning policies. The reward is binary since the agent is either correct or incorrect.

For CivilComments-WILDS (i.e., **Civil**), we study toxicity classification under subpopulation distribution shift. Each comment is annotated with the demographic identities it mentions, and toxicity classifiers can exhibit different error patterns across these identity subpopulations. We use pretrained LLM inference results from the HELM database Liang et al. (2022) and construct the task context by applying PCA to sentence embeddings, retaining the first 10 principal components. We consider two model pairs exhibiting heterogeneous expertise across demographic subpopulations. We use `together-redpajama-incite-instruct-3b-v1` and `meta-llama-7b`. The relative accuracy of the two agents reverses across the corresponding subpopulations. The reward is binary, indicating whether the assigned model correctly predicts the toxicity label.

For **ImageNet16H**, we use a VGG-19 Convolutional Neural Network fine-tuned for 10 epochs as one arm, and the predictions of Human 48 as the other. This is a multi-class classification problem with 16 classes. We assume symmetric misclassification utilities, which allows us to use a binary reward signal for our policies. We use *phase noise*, a feature in the dataset, as our singular context feature.

We train two linear classifiers as the two arms for **Camelyon17**, using data from two different hospitals to induce distribution shift and heterogeneous expertise. The binary classification task is detecting the presence/absence of a tumor. We extract 21 features from the pixels in the raw image. We identify the first principal component and use that as context in our model. The features used for the **Camelyon17** dataset are as follows:

1. Mean pixel intensity (R channel)
2. Standard deviation of pixel intensity (R channel)
3. 10th percentile of pixel intensity (R channel)
4. 90th percentile of pixel intensity (R channel)
5. Maximum pixel intensity (R channel)
6. Mean pixel intensity (G channel)
7. Standard deviation of pixel intensity (G channel)
8. 10th percentile of pixel intensity (G channel)
9. 90th percentile of pixel intensity (G channel)
10. Maximum pixel intensity (G channel)
11. Mean pixel intensity (B channel)
12. Standard deviation of pixel intensity (B channel)

13. 10th percentile of pixel intensity (B channel)
14. 90th percentile of pixel intensity (B channel)
15. Maximum pixel intensity (B channel)
16. Whiteness fraction (proportion of pixels with R, G, B > 220; proxy for background)
17. A proxy for Hematoxylin: mean of $B - R$ (purpleness; indicative of nuclear staining)
18. A proxy for Eosin: mean of $R - B$ (pinkness; indicative of cytoplasm staining)
19. Mean intensity across all channels
20. Colour contrast: standard deviation of grayscale intensity
21. Mean G channel intensity

For four tabular datasets (**Bank**, **Credit**, **Coupon**, **Cardio**), we construct two classifier agents as two arms. Agent 1 is a logistic regression model trained on a randomly selected subset of up to 20% of the available features, while Agent 2 is an XGBoost classifier trained on the remaining features. This asymmetric feature access and model class induce differences in marginal accuracy and context-dependent expertise. The contextual assignment policy observes the full feature space and uses observed rewards to learn which agent is better suited for each context. For the logistic regression agent, we use the `lbfgs` solver with balanced class weights and a maximum of 1000 iterations. For the XGBoost agent, we use 200 trees with maximum depth 4, learning rate 0.05, subsampling rate 0.8, column subsampling rate 0.8, and log-loss evaluation, with the random seed fixed to the experiment seed.

H.2 Additional Dataset Results

To conserve space in the main text, we report the full capacity-sweep results here for the remaining four datasets introduced in Section 4.1: three tabular datasets, **Credit** (Yeh & Lien, 2009) (23 features, 9,000 tasks, predicting default), **Coupon** (Wang et al., 2017) (24 features, 3,624 tasks, predicting coupon acceptance), and **Cardio** (11 features, 21,000 tasks, predicting cardiovascular disease); and one text-based dataset, **Civil**, a toxicity classification benchmark with demographic subpopulation shift. These follow the same experimental protocol as Section 4.1, see Appendix H.1 for details.

Figure 5 reports the average error rate as a function of Agent 1's capacity α_1 for all four datasets. As in the main text, contextual policies (logistic and tree-based, both Thompson Sampling and Greedy) consistently outperform the non-contextual baseline across all capacity levels, tree-based models generally outperform logistic models, and Thompson Sampling and Greedy policies perform similarly, consistent with the implicit exploration induced by the capacity constraints discussed in Section 4.1.

H.3 Uncertainty Across Randomized Task Streams

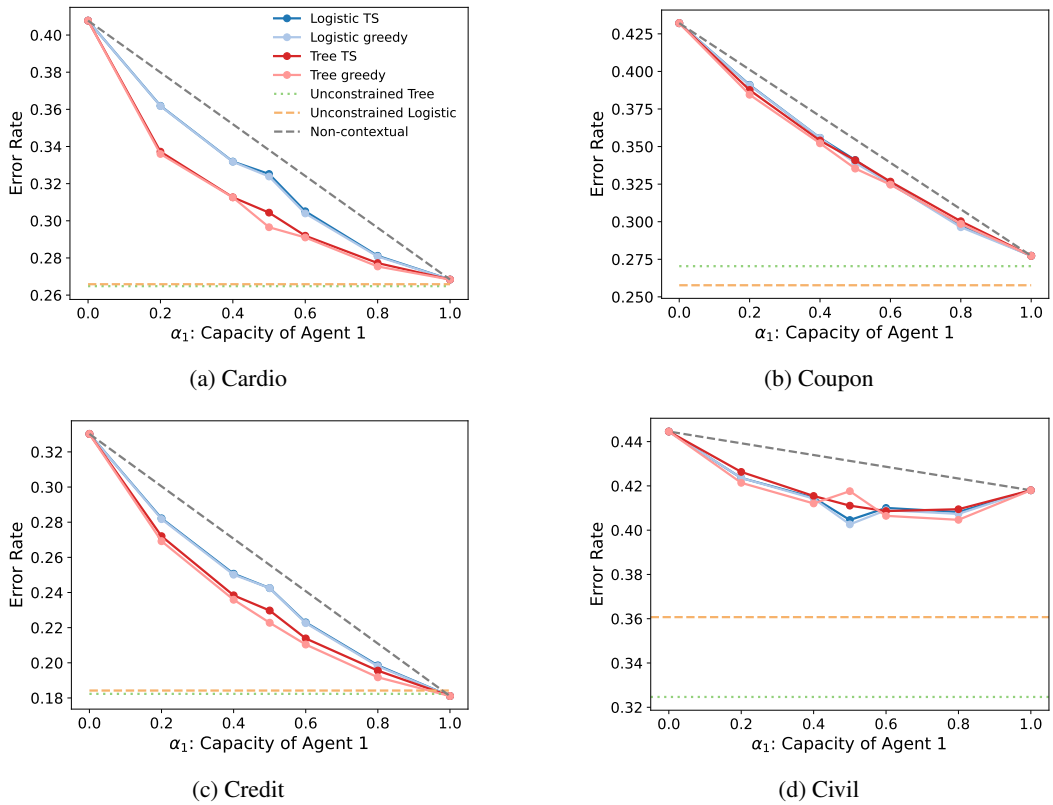
The main results in Figures 2b, 3, 5 are averaged over 100 randomized permutations of the online task sequence. Table 2 summarizes uncertainty across these runs. For each dataset, we report the range of error reductions relative to the non-contextual baseline across the four contextual policies and five evaluated capacity levels, together with the largest standard error of the mean error rate across these settings.

Across all datasets, the standard errors are small relative to the observed improvements over non-contextual assignment, indicating that the reported gains are stable across randomized task orderings.

H.4 Additional Results: Tabular Datasets Across Random Seeds

We provide additional robustness checks using multiple random seeds. We use the same agent-construction procedure as in Section 4.1: for each dataset, one agent is a logistic regression model trained on a random subset of up to 20% of the features, while the other is an XGBoost classifier trained on the remaining features.

While Figure 3 reports results for one seed, Table 3 reports results across additional seeds for **Bank**, **Credit**, **Coupon**, and **Cardio**. Across seeds, online contextual policies consistently outperform the



(a) Cardio

(b) Coupon

(c) Credit

(d) Civil

Figure 5: Error rate as a function of capacity (two-agent case), for the four additional datasets not shown in Figure 3. Points represent the average error rate over 100 runs. Uncertainty estimates are reported in Appendix H.3.

Table 2: Uncertainty across randomized task streams. Error reductions and standard errors are reported in percentage points. Each row summarizes four contextual policies and five evaluated capacity levels over 100 randomized task streams.

Dataset	Error reduction vs. non-contextual	Max. SE
Bank	4.80–10.70	0.031
Credit	1.30–4.40	0.070
Coupon	0.60–1.80	0.064
Cardio	1.30–4.40	0.070
MMLU	2.47–6.15	0.290
ImageNet16H	1.26–4.21	0.090
Camelyon17	4.25–8.03	0.080
Civil	4.07–6.65	0.001

non-contextual baseline. Tree-based greedy policies generally achieve comparable or lower error than logistic greedy policies. The magnitude of the improvement varies across seeds because the induced agent expertise depends on the random feature split; when the two agents have more similar expertise, the gains from contextual assignment are smaller.

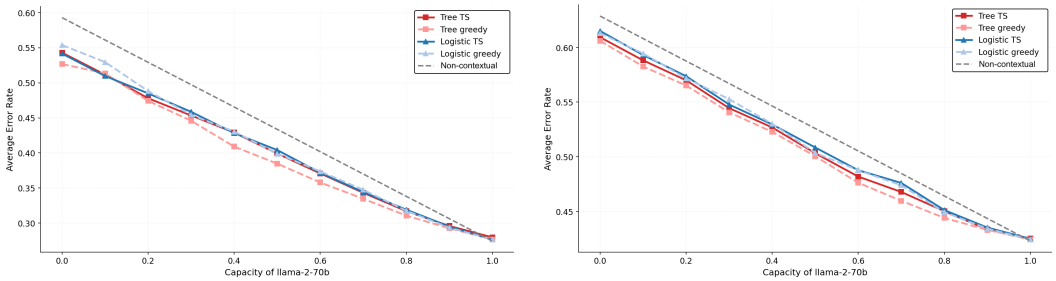
H.5 Additional Results: Multi-Agent Experiments with Capacity Constraints

We extend the two-agent setting to multi-agent assignment and evaluate the proposed policies on the **MMLU** dataset. Specifically, we consider settings with $A = 3$ agents in Figure 6a and $A = 5$ agents in Figure 6b. In each experiment, we vary the capacity assigned to one agent, LLaMa-2-70B, and allocate the remaining capacity equally across the other agents.

Table 3: Error rates across datasets and seeds, average over 10 runs, for fixed capacity with two agents $\alpha 1 = \alpha 2 = 0.5$

Dataset	Seed	Non-contextual	Logistic Greedy	Tree Greedy
Bank	0	0.128	0.122	0.113
	1	0.136	0.129	0.117
	2	0.278	0.195	0.171
	3	0.260	0.187	0.180
	4	0.124	0.097	0.097
Credit	0	0.195	0.190	0.188
	1	0.194	0.190	0.188
	2	0.256	0.241	0.222
	3	0.257	0.242	0.224
	4	0.197	0.192	0.188
Coupon	0	0.352	0.343	0.342
	1	0.369	0.357	0.360
	2	0.357	0.335	0.341
	3	0.347	0.331	0.340
	4	0.349	0.341	0.345
Cardio	0	0.334	0.332	0.309
	1	0.373	0.354	0.309
	2	0.338	0.335	0.306
	3	0.340	0.338	0.307
	4	0.343	0.338	0.311

Across both multi-agent settings, the contextual policies consistently achieve lower average error rates than the non-contextual assignment baseline. These results show that the benefits of contextual assignment are not limited to the two-agent case: when capacity must be allocated across multiple agents, learning context-dependent expertise still enables the policy to better match tasks to agents while respecting capacity constraints. This provides additional empirical support for the theoretical analysis in Section 2.3 and the main results in Section 4.1.



(a) (b)

Figure 6: Multi-agent assignment results on **MMLU**. (a) evaluates three LLM agents (Cohere Small (20220720), LLaMa-13B, and LLaMa-2-70B) on College Chemistry, Computer Security, and US Foreign Policy. (b) evaluates five LLM agents (Cohere Small (20220720), LLaMa-13B, LLaMa-2-70B, command-medium, and t0pp) on Abstract Algebra, College Chemistry, Computer Security, Econometrics, and US Foreign Policy. Results are averaged over 25 runs.

H.6 Additional Results: Extensions with an Additional Unconstrained Agent

In some settings, only a subset of agents face meaningful capacity constraints. For example, in a hospital setting, two doctors may have limited bandwidth, while a machine learning model can process cases at negligible marginal cost. To capture this setting, we consider an extension with two

constrained agents and one unconstrained "free" agent. Let agents $a \in \{1, 2\}$ be capacity-constrained, with target capacities α_1 and α_2, where $\alpha_1 + \alpha_2 = 1$, and let agent $a = 3$ denote the unconstrained agent. The online assignment rule remains as 2 that

$$a_t = \arg \max_{a \in \{1,2,3\}} \{\mu_{a,t}(x_t) - \eta Q_{t,a}\},$$

where $\mu_{a,t}(x_t)$ is the current estimate of agent a's expected reward for context x_t. For the constrained agents, the queues evolve as

$$Q_{t+1,a} = \left[Q_{t,a} + \mathbb{1}\{a_t = a\} - \alpha_a\right]_+, \qquad a \in \{1, 2\}.$$

The unconstrained agent has no capacity penalty, which is equivalent to setting $Q_{t,3} = 0 \quad$ for all t. Thus, the free agent enters the same assignment rule as the constrained agents, but without a queue penalty (i.e., $a_t = \arg\max_{a \in \{1,2,3\}} \{\mu_{a,t}(x_t) - \eta Q_{t,a}\}, Q_{t,3} = 0$).

In our implementation, the constrained agents are updated exactly as in the main setting, and the free agent's reward model is updated whenever it is selected. The capacity queues are updated only for the constrained agents. This design allows the policy to route tasks either to capacity-limited agents or to the unconstrained model, while ensuring that the constrained agents do not exceed their target capacities.

We evaluate the extension using the **Bank** dataset and the same two constrained agents as in Section 4.1. We add one unconstrained "free agent" with a marginal error rate 0.30. Results are averaged over 50 runs with $\eta = 0.5$. Figure 7 reports the average error rate as the capacity of Agent 1 varies from 0 to 1. All contextual policies outperform both the always querying free-agent baseline and the two-agent non-contextual random baseline. This shows that the unconstrained agent can improve performance by handling tasks for which the capacity-constrained agents have low adjusted scores (i.e., $\mu_{a,t}(x_t) - \eta Q_{t,a}$).

Unlike the two-agent setting, the endpoints do not correspond to the marginal error rate of a single constrained agent. When one constrained agent has zero capacity, the policy still chooses between the other constrained agent and the free agent. The policies also do not converge to the same endpoint performance, since different learning rules estimate agent expertise differently.

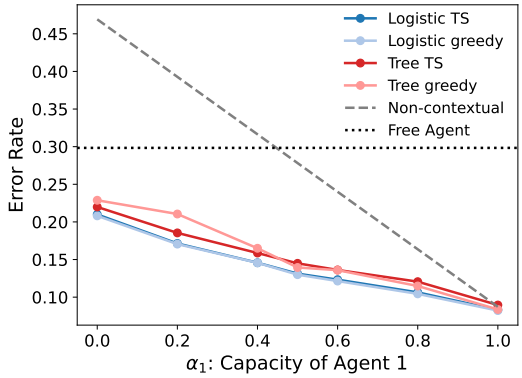


Figure 7: Extension with an unconstrained "free agent" and two capacity-constrained agents on the **Bank** dataset. The figure reports average error rates over 50 runs as the capacity of Agent 1 varies.

H.7 Additional Results: Mini-Batch Setting

The mini-batch setting differs from the main online setting only in how assignments and updates are grouped. Instead of assigning one task at a time, tasks are processed in batches of size B. For each batch, we compute estimated rewards $\mu_{a,t}(x)$ for every agent-task pair. A subtle distinction from the fully online setting is that batch-level assignments must use integer counts. Given target capacities $\{\alpha_a\}$, we first determine the number of tasks assigned to each agent in the current batch. These batch-level counts are chosen to approximate the target capacities while accounting for accumulated queue imbalances from previous batches. For example, if $B = 11$ and $\alpha_1 = \alpha_2 = 0.5$, one agent

must receive 6 tasks and the other 5 tasks in that batch; the virtual queue tracks this discrepancy so that future batches can compensate.

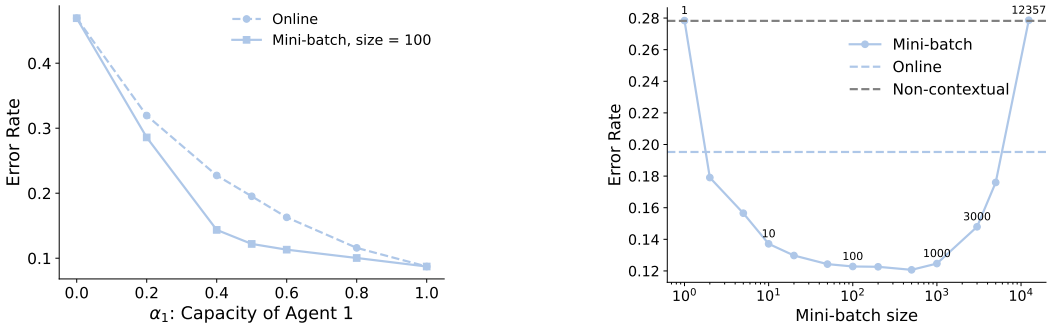
Conditional on these batch-level counts, we solve the assignment problem using a min-cost max-flow (MCMF) algorithm with adjusted scores $\mu_{a,t}(x) - \eta Q_{t,a}$. The MCMF step assigns tasks to agents to maximize predicted reward subject to the hard batch-level count constraints. Because $Q_{t,a}$ is constant across tasks within a batch for a fixed agent, the queue term does not affect the within-batch matching conditional on fixed counts. Its main role is to adjust future batch counts when exact proportional allocation is not possible within a batch.

After the batch is assigned, rewards are observed, the reward models are updated using the observed outcomes, and the virtual queues are updated based on the number of tasks assigned to each agent in the batch:

$$Q_{t+B,a} = [Q_{t,a} + N_{t,a} - B\alpha_a]_+ ,$$

where $N_{t,a}$ is the number of tasks assigned to agent a in the batch. Thus, capacity is enforced as a hard constraint within each batch through integer assignment counts, while the queue maintains long-run capacity balance across batches.

In addition to the tree-based greedy results reported in Figure 4, we report the corresponding logistic greedy results in Figure 8. The patterns are consistent: mini-batch assignment with $B = 100$ improves over the fully online setting by enabling joint allocation over multiple tasks. Varying the batch size again reveals a U-shaped pattern: very small batches provide frequent updates but less stable capacity allocation, while very large batches delay feedback and slow learning. Intermediate batch sizes achieve the lowest error. These improvements, however, come at the cost of increased latency.



(a) Online vs mini-batch assignment with batch size $N_B = 100$. Error rates are averaged over 50 runs as the capacity of Agent 1 varies.

(b) Effect of batch size when Agents 1 and 2 have equal capacities, $\alpha_1 = \alpha_2 = 0.5$. Average error rates are computed over 10 runs.

Figure 8: Mini-batch assignment on the **Bank** dataset using the logistic greedy policy.

H.8 Additional Results: Effect of the Queue Penalty Parameter η

As discussed in Section 3 and Appendix F, the queue penalty parameter η controls the trade-off between reward maximization and capacity enforcement. The online policy selects

$$a_t = \arg \max_{a \in \{1,\dots,A\}} \{\mu_{a,t}(x_t) - \eta Q_{t,a}\} ,$$

as in equation 2, where $\mu_{a,t}(x_t)$ is the current estimate of agent a's expected reward for context x_t and $Q_{t,a}$ is the virtual queue tracking capacity pressure. Smaller values of η place more weight on estimated expertise, while larger values place more weight on satisfying the target capacities.

For all experiments in Section 4 and Appendix H.2, we fix $\eta = 0.5$. Here, we examine the sensitivity of performance to this choice. We use the **Camelyon17** setting from Section 4.1, vary the capacity of Agent 1, and compare tree-based greedy policies with different values of η against the non-contextual random baseline. Results are averaged over 100 runs. Figure 9 shows the results. As expected, smaller values of η lead to lower error rates because the assignment rule places more weight on estimated

agent expertise and allows short-run deviations from the target capacities. When $\eta = 0$, the policy converges to the unconstrained contextual policy, selecting agents purely based on estimated reward. As η increases, the queue penalty plays a larger role in assignment decisions. For large values such as $\eta = 5$, the capacity term dominates, and performance approaches the non-contextual random baseline because assignments are driven primarily by capacity balancing rather than contextual expertise.

Overall, η determines how strongly the decision-maker prioritizes capacity enforcement relative to predictive performance. The appropriate choice depends on the application: settings with strict workload or fairness requirements may prefer larger η, while settings that tolerate short-run capacity deviations may benefit from smaller η.

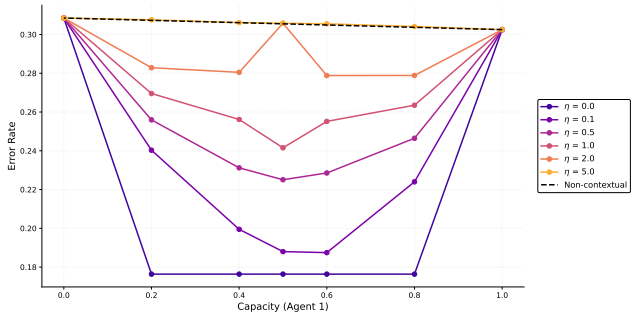


Figure 9: Effect of the queue penalty parameter η in the **Camelyon** dataset (described in Section 4.1. Smaller η prioritizes reward maximization and approaches the unconstrained contextual policy, while larger η enforces capacity more aggressively and approaches the non-contextual random baseline.

H.9 Validation of Capacity Constraint Enforcement

We validate the queue-based capacity mechanism in two complementary settings: a controlled simulation where the optimal constrained allocation and shadow price are analytically known, and a real-data experiment using the Bank dataset.

Synthetic validation. We first consider a setting with known agent expertise. We draw $x \sim \mathcal{N}(0, 1)$ and set

$$\mu_1(x) = \sigma(2x), \qquad \mu_2(x) = \sigma(-2x),$$

with target capacities $(\alpha_1, \alpha_2) = (0.3, 0.7)$ and $\eta = 0.5$. We run 50 independent task streams of $T = 5{,}000$ tasks each. Because $\mu_1(x) - \mu_2(x)$ is increasing in x, the optimal constrained policy assigns Agent 1 to the largest 30% of contexts. The corresponding threshold is

$$P(X \geq x^\star) = 0.3 \quad \Longrightarrow \quad x^\star = \Phi^{-1}(0.7) = 0.524.$$

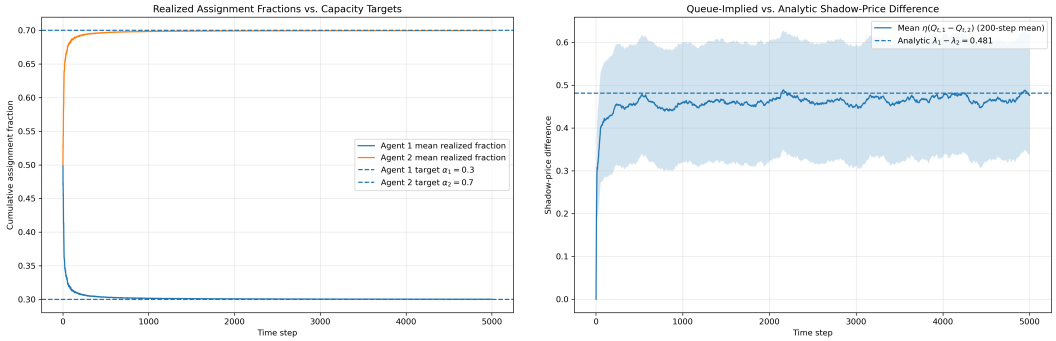
At this threshold, the analytical shadow-price difference is

$$\lambda_1 - \lambda_2 = \mu_1(x^\star) - \mu_2(x^\star) = 0.481.$$

Figure 10 shows that the online queue mechanism closely reproduces both analytical predictions. The mean final realized assignment fractions are 0.300 and 0.700, matching the target capacities. Moreover, the queue-implied shadow-price difference, $\eta(Q_{t,1} - Q_{t,2})$, averages 0.471 over the final 500 tasks, close to the analytical value 0.481.

Real-data validation. We further evaluate capacity enforcement on the Bank dataset using target capacities $(\alpha_1, \alpha_2) = (0.2, 0.8)$ and $\eta = 0.5$ across 100 randomized task streams. We define convergence as the first task at which the cumulative realized assignment fraction is within 0.01 of the target capacity and remains within this tolerance for the subsequent 500 tasks.

As shown in Table 4, all four policies satisfy this criterion in all 100 runs, and none subsequently violates the 0.01 tolerance. Median convergence occurs within the first 158 tasks for every policy, or at most approximately 1.3% of the task horizon. The maximum final absolute deviation from the target capacity across all runs is only 0.000129.



(a) Realized assignment fractions versus capacity targets.

(b) Queue-implied versus analytical shadow-price difference.

Figure 10: Synthetic validation of the queue-based capacity mechanism. Solid curves show means across 50 randomized task streams and shaded regions show 95% confidence intervals. Dashed lines denote the analytical targets. The shadow-price trajectory in panel (b) uses a 200-step rolling mean.

Table 4: Capacity enforcement on the Bank dataset across 100 randomized task streams. Convergence is defined as entering a ± 0.01 neighborhood of the target capacity and remaining within it for the subsequent 500 tasks.

Policy	Median conv.	Mean conv.	Mean conv. (% T)
Logistic TS	158	156	1.26%
Tree TS	153	156	1.26%
Logistic Greedy	132	128	1.04%
Tree Greedy	96	103	0.84%

We repeat this evaluation on all other datasets under the same target capacities and η. Table 5 reports, for all experiments, the worst case across the four contextual policies of the realized deviation from the target capacity, averaged over the final 5% of each horizon. The deviation is at most 0.0113 across every dataset and policy, and below 0.001 for all datasets except MMLU, whose much shorter horizon ($T = 217$, versus $T \geq 2{,}000$ for the others) leaves less room for the running assignment fraction to average out. Together with the synthetic results above, these results show that the virtual queue mechanism closely enforces the desired capacity targets across experiments.

H.10 Robustness to the Feature-Split Choice

The tabular experiments in Section 4.1 construct two agents by randomly partitioning each dataset's features: Agent 1 is a logistic regression model trained on a subset of up to 20% of the features, and Agent 2 is an XGBoost classifier trained on the remainder. To verify that the measured gains of contextual assignment over the non-contextual (random) baseline are not an artifact of this particular 20/80 split, we sweep the fraction of features assigned to Agent 1 over $\{0.1, 0.2, 0.5, 0.8, 0.9\}$ and, for each fraction, over 5 independent random seeds governing which specific features are assigned to which agent, yielding 25 distinct agent configurations on the **Bank** dataset. For each configuration we evaluate all four contextual policies (Logistic TS, Tree TS, Logistic Greedy, Tree Greedy) at $\alpha_1 = \alpha_2 = 0.5$ and $\eta = 0.5$, and report the gain over random assignment.

Table 6 reports the mean, minimum, and maximum gain over random assignment across 5 seeds and 4 policies for each split fraction (20 evaluations per row). The gain is positive in all 100 evaluations, confirming that the improvement is not specific to the 20/80 split used in Section 4.1. Consistent with Appendix D.4, the gain correlates with the realized disagreement mass $p_1 + p_2$ at $r = 0.80$–0.87 across policies, suggesting that the split fraction affects the gain through the disagreement it induces.

Table 5: Capacity enforcement across datasets, reporting the worst case over the four contextual policies (Logistic TS, Tree TS, Logistic Greedy, Tree Greedy). We report the mean and maximum, across 100 randomized task streams, of the absolute deviation between the realized assignment fraction and the target capacity $\alpha_1 = 0.2$, averaged over the final 5% of each dataset's horizon.

Dataset	Mean final deviation	Max final deviation
Bank	0.0001	0.0001
Cardio	0.0001	0.0001
Credit	0.0002	0.0002
Coupon	0.0004	0.0005
Camelyon17	0.0009	0.0011
ImageNet16H	0.0006	0.0009
Civil	0.0003	0.0004
MMLU	0.0072	0.0113

Table 6: Gain over the capacity-matched random baseline as a function of the feature-split fraction assigned to Agent 1, on the **Bank** dataset. Each row aggregates 5 feature-assignment seeds $\times$ 4 contextual policies (20 evaluations) at $\alpha_1 = \alpha_2 = 0.5$, $\eta = 0.5$.

Split (α_1 features to Agent 1)	Mean gain	Min gain	Max gain
0.1	0.0301	0.0045	0.0470
0.2	0.0429	0.0050	0.1072
0.5	0.0329	0.0051	0.0871
0.8	0.0096	0.0023	0.0172
0.9	0.0086	0.0027	0.0142