Optimal Auditing for Alignment: An Audit Index for Label Allocation under Reference-Conditioned Preference Optimization

Liz Lemma Future Detective

January 22, 2026

Abstract

Preference optimization methods such as DPO and RLHF use a reference model to regularize updates, but recent evidence shows they rarely correct preference mis-rankings inherited from the reference model and can exhibit a growing gap between off-policy ranking metrics and on-policy win rate. In parallel, scalable oversight work shows that weak judges can be easily persuaded in open consultancy, risking mistake amplification, while adversarial protocols such as debate reduce amplification. We bring these threads together by modeling alignment as an auditing and resource allocation problem: given limited labeling capacity and compliance constraints typical of 2026 deployment pipelines, which preference comparisons should be collected and recollected? Using the idealized DPO/RLHF characterization of the optimal likelihood ratio (Chen et al., 2024), we derive a tractable condition under which a labeled datapoint can be correctly ranked: $\alpha > \sigma(\beta c)$, where c is the reference log-ratio favoring the rejected output. Under a Beta–Binomial model of rater noise, the expected welfare gain from labeling a datapoint becomes a closed-form Beta tail probability. This yields an 'alignment audit index' that prioritizes datapoints with large reference error magnitudes (high c) and high posterior uncertainty about the threshold crossing (disagreement), and supports greedy/index policies with near-optimality guarantees under concavity/submodularity conditions. We propose practical estimators of c from model log-probabilities and uncertainty from repeated ratings or proxy judges, and outline empirical simulations on preference datasets to show that index-based auditing dominates uniform sampling in both ranking correction and win rate improvements for a fixed labeling budget.

Table of Contents

(consultancy/debate) to a resource allocation perspective.

2. 2. Empirical stylized facts to match: (i) ranking flips are rare under DPO before early stopping; (ii) difficulty depends on reference log-ratio; (iii) weak judges can amplify mistakes; motivate targeted auditing rather than uniform data collection.

3. 3. Setup and primitives: population of datapoints with c_i, α_i, weights v_i; labeling budget K; Beta–Binomial noise; deployment welfare definition.

4. 4. Idealized learnability and the threshold form: derive $\alpha > \sigma(\beta c)$ from Chen et al. and restate idealized ranking accuracy in threshold notation; discuss interpretation of β as inertia/adjustment cost.

5. 5. The audit index: closed-form posterior tail probability $p_i(n_i) = \Pr(\alpha_i > \sigma(\beta c_i) \mid \mathcal{D}_i)$; define I_i and marginal value Δ_i; show how disagreement enters through posterior mass near the threshold.

6. 6. Allocation problem and solution concepts: knapsack vs sequential (bandit) label allocation; relaxed KKT characterization; greedy/index policy; conditions for monotonicity and diminishing returns.

7. 7. Near-optimality results: submodularity/concavity sufficient conditions; approximation guarantee for greedy; discussion of when numerical methods (dynamic programming / Bayesian bandits) are required.

8. 8. Comparative statics and design implications: how β, strength of π_{ref}, and rater noise shift the index and the optimal allocation; link to iterative/on-policy refresh and to oversight protocol choice.

9. 9. Empirical evaluation plan: simulate auditing on existing preference datasets (e.g., AlpacaFarm cross-annotations) using estimated c and observed disagreement; compare uniform vs index-based acquisition for fixed K; measure ranking flips and win rate; sensitivity to reference quality and β.

10. 10. Discussion: governance/auditability in 2026; what regulators or internal assurance teams can compute and report (c-distribution, audited coverage, expected correction mass); limitations and extensions (multi-output ranking, heterogeneous raters, strategic judges).

1 Introduction and motivation: alignment as auditing under scarce labels

Much of contemporary "alignment work" can be reinterpreted as a problem of *auditing*: we deploy (or plan to deploy) a powerful policy, we suspect that some portions of its behavior are misaligned with the relevant normative or task objective, and we have a limited budget of human attention to detect and correct those failures. The basic friction is not that we cannot elicit *any* feedback, but that we cannot elicit *enough* feedback to comprehensively supervise all behaviors that matter in deployment. In practice, labeling budgets are scarce for mundane reasons (time, money, evaluator availability, and the cost of domain expertise), and scarce for deeper reasons (some evaluations are intrinsically slow, adversarially hard, or require rare contextual knowledge). If we treat human feedback as the primary corrective signal, then alignment becomes a resource-allocation problem: how should we spend limited comparisons, demonstrations, or audits so as to maximize safety-relevant performance where it matters most?

This auditing perspective is especially natural under modern preference-optimization pipelines. In RLHF-style systems, and in direct preference optimization (DPO) variants, the learning signal is not an absolute "gold" label but a collection of *relative* judgments: given a prompt (or state), which of two candidate outputs is better? These judgments are noisy, costly, and heterogeneous in difficulty. Moreover, the downstream optimizer does not treat the dataset as a passive record; it treats it as evidence that is combined with a *reference policy* through an explicit regularization mechanism (e.g. a KL penalty or an equivalent reference-conditioning). As a result, the training outcome is shaped by two interacting forces: (i) what humans would prefer in principle, and (ii) how strongly the training procedure is willing to deviate from the reference model to satisfy those preferences. From an auditing standpoint, this means that not all datapoints are equally "actionable" given a fixed training recipe: some comparisons are easy for the optimizer to incorporate because they ask for modest deviations from the reference, while others require substantial movement against the reference model's implicit prior.

We can motivate this interaction without committing to any particular algorithmic details. A reference-conditioned optimizer (whether realized as KL-regularized RL, DPO, or a close cousin) implicitly implements a kind of *inertia*: it prefers changes that do not move too far from what the reference already assigns high probability. When a proposed improvement corresponds to a response that the reference already finds plausible, the optimizer can readily amplify it; when the improvement corresponds to a response that the reference strongly disfavors, the optimizer must pay a higher "deviation cost." This observation is well-known informally (practitioners talk about

certain edits being "too far" from the base model), but it has a consequential implication for auditing: the *value* of collecting more labels on a datapoint depends not only on how important that datapoint is in deployment, but also on where it lies relative to the reference model's inductive bias and the regularization strength.

We also emphasize that "auditing" here is not merely post hoc evaluation. In a safety-oriented training loop, auditing is a governance-relevant mechanism: it is how a principal (the lab, regulator, or deployment owner) decides what evidence to purchase before committing to a model update or a release. In that role, audits must be prioritized. A lab that spends its evaluation budget uniformly across prompts is implicitly asserting that all prompts are equally likely to be decisive for safety—an assumption that is rarely defensible. Conversely, targeted auditing can be understood as a disciplined attempt to build a *safety case* with limited evidence: we want to concentrate evaluator effort on the parts of behavior where (a) failures would be costly, and (b) the available training update is plausibly capable of fixing them.

The scarcity of reliable labels becomes even more salient when we move beyond "strong supervision" and consider *weak oversight* schemes. In debate, consultancy, recursive reward modeling, or other scalable oversight paradigms, the evaluator is intentionally weaker than the system being trained. Weak oversight is attractive because it promises to amortize expensive expertise, but it introduces a second bottleneck: even if we can afford many evaluations, those evaluations may not track the intended objective on hard instances. Put differently, we are not only label-budget constrained; we are also *judge-quality constrained*. In reduced form, weak oversight can be modeled as an increase in label noise or as a systematic bias in what is judged "better." This again suggests an auditing lens: when judges are weak, we must be selective about which comparisons we buy, because some comparisons will predictably elicit unreliable judgments (or judgments that are manipulable by the model), while other comparisons remain stable and informative.

A central motivation for our formalization is therefore to connect three practical facts that are often discussed separately:

(i) Reference-conditioning creates a learnability boundary. Even with perfect labels, a heavily regularized optimizer may not move on certain datapoints, because doing so would require large deviations from the reference. In the language of DPO/RLHF, the regularization parameter (often expressed as a KL weight or as an equivalent β) mediates the tradeoff between fitting preferences and staying close to π_{ref}. The same dataset can lead to qualitatively different outcomes depending on this tradeoff. From a safety perspective, this means that collecting additional labels on some

"hard" comparisons may be wasted effort unless we also change the training recipe; and conversely, for "nearby" improvements, relatively few labels may suffice.

(ii) Oversight is heterogeneous in difficulty and importance. A preference comparison is not a monolith. Some prompts correspond to frequent user requests or high-stakes domains (medical advice, cybersecurity, legal compliance), while others are rare or low consequence. Some comparisons are easy for raters (e.g. clear toxicity) and others require nuanced reasoning (e.g. subtle deception, long-horizon consequences, or domain expertise). Treating all comparisons as exchangeable ignores this heterogeneity and can misallocate evaluator effort away from the highest-welfare opportunities.

(iii) Weak judges can amplify mistakes rather than correct them. In weak oversight settings, a model can sometimes exploit evaluator blind spots, producing outputs that appear good under the judge but are actually undesirable. If the training loop then optimizes against that flawed signal, it can entrench or amplify misbehavior. Importantly, this failure mode interacts with reference-conditioning: a reference model may already be biased toward outputs that are persuasive-but-wrong, and regularized optimization can preserve that bias unless the evidence is strong enough to overcome it. In this sense, "better auditing" is not just about collecting more labels; it is about choosing labels that reduce the probability of crossing the wrong decision boundary.

These considerations motivate a resource-allocation viewpoint: we want to decide *where* to spend limited comparisons to improve the probability that training actually changes the model's ranking in the desirable direction on the most important behaviors. The key move in our approach is to treat each datapoint as an "audit target" with an associated (i) deployment importance, (ii) reference-model difficulty, and (iii) uncertainty about true human preference. This is the intuitive content behind the objects introduced in the enclosing scope: we observe a reference log-ratio summarizing how much the reference model prefers one candidate over another, we posit an underlying preference probability capturing rater agreement, and we interpret the training step as succeeding on a datapoint if (and only if) the latent preference signal is strong enough to overcome the reference-conditioned inertia.

This reframing also clarifies what is and is not being claimed. We are not assuming that the world decomposes neatly into independent preference pairs, nor that training literally flips a binary switch per datapoint. Rather, we use a simplified criterion because it exposes the safety tradeoff we care about: under reference-conditioning, there is a region where additional evidence changes our beliefs but does not change the optimizer's effective deci-

sion, and a region where evidence is decisive for whether training will correct a ranking. Auditing should concentrate on the boundary between these regions, especially when the boundary lies in high-welfare parts of the input space. In other words, if alignment is about reducing the probability of catastrophic or high-cost failures under constrained oversight, then the principal's problem is structurally similar to optimal testing: allocate experiments to where the expected value of information (for downstream action) is largest.

Finally, this perspective connects to governance and verification. External auditors and internal red teams already face a version of the label allocation problem: they cannot test every behavior, so they choose test suites, adversarial prompts, and evaluation protocols. Our contribution is to tie that selection problem to the *training* mechanism itself. If the deployment owner knows that certain failures are "far" from the reference model under a given β, then an audit that only measures failure frequency is incomplete; one also wants to measure whether failures are *correctable* by the planned update. Conversely, if failures are near the reference boundary and raters are uncertain, targeted additional labeling can be particularly valuable. This yields a concrete, operational message: safety evaluations should be coupled to an explicit model of how evidence translates into training changes, rather than being treated as a separate, purely descriptive layer.

In the next section, we translate these intuitions into empirical desiderata. We will ask for a stylized model consistent with observed behavior of preference-trained systems: that ranking flips are not ubiquitous, that difficulty tracks the reference log-ratio in a systematic way, and that weak judges can systematically mislead training. These facts, taken together, push strongly toward targeted auditing and away from uniform data collection as a default.

2 Empirical stylized facts: when labels change the model (and when they do not)

Our allocation problem is only worth formalizing if it captures something stable about how preference-trained systems actually behave. In this section we summarize three empirical stylized facts—all familiar to practitioners in one form or another—that jointly motivate treating "where to label" as a first-class design choice rather than an afterthought. The common theme is that preference optimization is *not* an unconstrained supervised learner: it is a regularized update around a reference, and thus it exhibits sharp heterogeneity in which comparisons are (i) learnable, (ii) important, and (iii) trustworthy under real oversight.

Fact 1: ranking flips are rare under DPO unless we push aggressively (or stop early for the wrong reason). In many DPO/RLHF

training runs, especially those tuned for stability, the post-training policy does not massively reorder preferences across the space of prompts. Instead, most comparisons remain essentially as they were under the reference policy: the model may become more consistent, less toxic, or more helpful in aggregate, but for a large fraction of specific prompt–response pairs, the *direction* of preference between two candidates is unchanged. This is visible operationally in several ways: win-rate improvements plateau quickly on in-distribution preference sets; many prompts show negligible change in pairwise log-odds; and fine-tuning often yields a small number of salient behavior shifts (e.g. refusal boundary movement) rather than pervasive re-ranking everywhere.

There are benign and non-benign interpretations of this. The benign story is simply that the reference model was already near a local optimum for the preference distribution we care about, and the update merely refines it. The non-benign story is that we are constrained by inertia: KL regularization, implicit trust-region behavior, and conservative hyperparameter choices can prevent learning on comparisons that would require moving "far" from the reference. In practice, teams often observe that making the update more aggressive—larger effective step sizes, lower regularization, more epochs—can increase measurable preference fitting but also increases instability: regressions on unrelated capabilities, reward hacking-like artifacts, or brittle behavior changes. These observations are consistent with a picture in which many potential "fixes" exist, but only a subset can be realized under a stable update.

From an auditing perspective, the key implication is that *additional labels do not uniformly translate into behavior change.* If most pairwise rankings would not flip under the intended training recipe, then auditing should focus on identifying (and resolving uncertainty around) the subset of cases where a flip is both feasible and consequential. Uniform data collection implicitly assumes the opposite: that every label has comparable marginal influence on the trained policy. The empirical reality is closer to a sparse-influence regime, where only a minority of comparisons are decision-relevant for the update.

Fact 2: "difficulty" is strongly mediated by the reference model's log-ratio. A second consistent observation is that the ease with which preference optimization can enforce a desired ranking correlates with how the reference model already scores the two candidates. When the reference already assigns comparable probability mass to the preferred output, training can amplify it with little resistance. When the preferred output is far into the reference tail, training must overcome a large log-probability gap, and doing so is either slow, unstable, or effectively prohibited under conservative regularization.

This dependence is not merely an artifact of optimization noise; it reflects a structural constraint. Under reference-conditioned objectives, the update is explicitly penalized for moving probability mass away from the reference. Thus, the "distance" to be traveled matters: pushing probability from a high-likelihood reference region into a low-likelihood region costs more than reshuffling mass within a locally plausible set. Practitioners often summarize this as: *it is easy to make the model prefer one good answer over another good answer; it is hard to make the model consistently produce an answer it initially considered implausible.*

Empirically, we can think of the reference log-ratio

$$c \;=\; \log \frac{\pi_{\mathrm{ref}}(y^{\ell} \mid x)}{\pi_{\mathrm{ref}}(y^{w} \mid x)}$$

as a coarse sufficient statistic for this notion of difficulty. Large positive c means the reference strongly prefers the "losing" candidate y^{ℓ} over the human-preferred y^{w}. These are exactly the cases where, absent very strong preference evidence and/or a weak regularizer, the optimizer tends to "give up" and preserve the reference ordering. Conversely, when c is small or negative, the preferred candidate is already plausible under the reference, and even modest evidence can tip the learned policy toward preferring it more robustly.

This fact has two immediate auditing consequences. First, if we are trying to discover correctable misalignment, we should expect it to cluster in regions where the reference is conflicted or only mildly wrong, not where it is massively confident in the wrong direction (unless we are willing to change the training recipe). Second, if we do want to target "deep" failures where the reference is confidently wrong, then data collection alone is unlikely to suffice; the audit should be coupled to an explicit intervention plan (e.g. revising β, adding demonstrations, changing the model class, or using a different update rule). Otherwise we risk spending evaluation budget to obtain evidence about failures that we cannot realistically fix in the current loop.

Fact 3: weak judges can systematically amplify mistakes rather than correct them. The third stylized fact concerns oversight quality. In real deployments, the principal rarely has access to perfect evaluators. Raters are time-constrained, may lack domain expertise, and can be influenced by presentation effects. When models become sophisticated, the evaluation problem can become adversarial: the model can produce outputs that are persuasive, fluent, and locally plausible, while being globally incorrect, manipulative, or strategically deceptive. In such regimes, preference labels are not merely noisy—they can be *biased* in a direction that the model can exploit.

This is visible in phenomena like sycophancy (outputs that match a user's stated beliefs rather than the truth), plausible-sounding hallucinations that fool non-expert raters, and "helpfulness" behaviors that trade off against safety in subtle ways. More generally, weak oversight creates a channel for Goodhart-like failures: optimizing a proxy (what judges select) can degrade the true objective (what we actually want), especially when the model can search over outputs that exploit judge blind spots. Importantly, this failure mode interacts with reference-conditioning. If the reference already leans toward the kinds of outputs that fool the judge, then a conservative update can preserve that bias. If the update is aggressive, it can amplify it by pushing probability mass toward the judge-fooling region even more strongly.

For auditing, the implication is not merely "collect more labels." If the marginal labels are coming from a systematically miscalibrated judge on certain categories of prompts, buying more of them can increase confidence in the *wrong* conclusion—and lead the principal to authorize a model update that entrenches misbehavior. Thus, the value of auditing is heterogeneous not only in (x, y^w, y^ℓ) but also in *oversight regime*: which prompts are susceptible to evaluator error, which are robust, and which are actively exploitable.

This suggests a richer notion of "high value of information." The valuable labels are those that (i) influence the downstream training decision and (ii) are trustworthy for the decision at hand. In practice, this pushes us toward targeted auditing strategies that emphasize prompts where evaluator reliability is high or can be made high via protocol design (expert review, adversarial testing, model-assisted critique, calibration tasks), and away from naively scaling weak labels on the hardest instances where raters are most easily misled.

Synthesis: why these facts point to targeted auditing. Taken together, these facts yield a coherent picture. Because ranking flips are rare, most additional labels have little effect on the trained policy's decisions; because difficulty depends on the reference log-ratio, the cases where the model *could* change are predictable from reference behavior; and because weak judges can introduce systematic bias, some labels are not merely low-value but actively harmful if treated as ground truth.

Uniform data collection ignores all three considerations. It spends the same marginal effort on (a) comparisons that the optimizer will not change under the planned recipe, (b) comparisons that are already settled under the reference and the human preference distribution, and (c) comparisons where judges are least reliable. A targeted auditing policy, by contrast, tries to concentrate effort on the boundary cases: prompts where the reference is wrong but not hopelessly so, where the preference signal is plausible but uncertain, where welfare stakes are large, and where oversight can be made sufficiently reliable to justify acting on the evidence.

This is also the point where governance considerations enter naturally. If a lab is building a safety case for a release, it is not enough to measure average preference win-rate; we need evidence that the update corrects high-stakes failures that are *actionable* under the intended training constraints. Targeted auditing can be understood as producing decision-relevant evidence: it prioritizes datapoints that are likely to change the release decision (or the training recipe) and deprioritizes those that are either already safe or currently unfixable.

In the remainder of the paper we translate this qualitative picture into a stylized but decision-theoretic model. The model will explicitly represent (i) a per-datapoint measure of reference difficulty (via a log-ratio), (ii) a latent preference probability that captures rater agreement and judge weakness in reduced form, and (iii) a welfare weight capturing deployment importance. With these primitives, the principal's problem becomes an allocation of a finite labeling budget across heterogeneous audit targets, where the marginal value of a label is highest precisely in the high-stakes, high-uncertainty region near the learnability boundary induced by reference-conditioning.

3 Setup and primitives

We now turn the qualitative auditing picture into a minimal decision-theoretic model. The goal is not to faithfully simulate modern RLHF pipelines end-to-end, but to isolate a few structural features that (i) are stable across implementations and (ii) are directly relevant to *where* additional preference labels change the deployed behavior. Concretely, we model a principal (the lab, or an internal safety team) that can purchase a finite number of pairwise preference comparisons and must decide how to allocate them across heterogeneous audit targets.

Datapoints and reference difficulty. We consider a finite population of pairwise preference datapoints indexed by $i \in \{1, \dots, N\}$. Each datapoint consists of a prompt x_i and two candidate completions (y_i^w, y_i^ℓ), where the superscripts are *ex post* labels indicating which completion is intended to be the "better" one under the target preference relation (e.g. more helpful, more honest, less harmful) and which is the "worse" one. We assume the principal can evaluate the *reference policy* π_{ref} on these candidates, yielding the observable reference log-ratio

$$c_i \;:=\; \log \frac{\pi_{\text{ref}}(y_i^\ell \mid x_i)}{\pi_{\text{ref}}(y_i^w \mid x_i)}. \tag{1}$$

Intuitively, c_i summarizes how strongly the reference model "disagrees" with the intended ordering: large positive c_i means π_{ref} assigns much higher probability to the dispreferred completion than to the preferred completion, while

$c_i \approx 0$ indicates that the reference treats the two candidates as similarly plausible. We take $\{c_i\}_{i=1}^N$ as free, observable heterogeneity that the principal can use *before* buying any labels.

Latent preference rates and rater noise. For each i, there is an unknown "true" preference probability $\alpha_i \in (0,1)$, which aggregates both human disagreement and judge weakness into a single reduced-form parameter:

$$\alpha_i \;:=\; \Pr\big(y_i^w \succ y_i^\ell \,\big|\, x_i\big). \tag{2}$$

Operationally, α_i is the probability that a randomly drawn rater (from the deployed oversight process) prefers y_i^w to y_i^ℓ when asked to compare them under the lab's rubric. We emphasize that α_i is not "ground truth" in a normative sense; it is the effective signal available to the training pipeline. When judges are weak or systematically biased on some class of prompts, that bias is expressed here as α_i taking values that do not match the principal's true desiderata. This is precisely why we treat the allocation of labels as a safety-relevant decision: buying more labels can either clarify correctable errors or concentrate confidence around an oversight failure.

Given α_i, each purchased comparison produces a binary outcome. If the principal buys n_i labels on datapoint i, we observe $w_i \in \{0, 1, \ldots, n_i\}$ "wins" for y_i^w, with likelihood

$$w_i \,\big|\, \alpha_i, n_i \;\sim\; \text{Binomial}(n_i, \alpha_i). \tag{3}$$

Equivalently, each individual label is an i.i.d. Bernoulli draw with success probability α_i. This is the standard Bradley–Terry/noisy-comparison abstraction; it is deliberately agnostic to the microfoundations of rater error, while still capturing the two audit-relevant dimensions: (i) the mean preference direction and (ii) the uncertainty remaining after finitely many comparisons.

Prior and posterior: conjugate auditing updates. We assume a conjugate prior $\alpha_i \sim \text{Beta}(a_0, b_0)$, either as a modeling choice or as a tractable approximation to a more complex empirical Bayes procedure. The Beta prior plays two roles. First, it provides a coherent way to translate finite-sample label counts into calibrated uncertainty, which matters because the value of additional labeling is driven by how often datapoints are near a decision boundary. Second, the hyperparameters (a_0, b_0) can be interpreted as encoding the principal's base-rate beliefs about rater agreement (e.g. whether most comparisons are "easy" with α_i near 0 or 1, or "hard" with α_i near 1/2).

Let $\mathcal{D}_i = (w_i, n_i)$ denote the label data purchased on datapoint i. By Beta–Binomial conjugacy, the posterior is

$$\alpha_i \,\big|\, \mathcal{D}_i \;\sim\; \text{Beta}(a_0 + w_i,\; b_0 + n_i - w_i). \tag{4}$$

We write $\hat{\alpha}_i := w_i/n_i$ for the empirical win rate (when $n_i > 0$), but emphasize that the posterior distribution, not just $\hat{\alpha}_i$, will drive the principal's allocation incentives: what matters is not only where $\hat{\alpha}_i$ lies, but also how uncertain we remain about α_i.

Deployment welfare weights. Not all datapoints matter equally. Some prompts occur frequently in deployment, some correspond to high-stakes decisions, and some are safety-critical even if rare. We encode this with an exogenous welfare weight $v_i \geq 0$. One interpretation is frequency weighting: v_i is proportional to the probability mass of the deployment distribution that falls into the "region" represented by datapoint i. Another is a governance-weighted value function: v_i incorporates externalities, tail risks, or regulatory constraints, so that a misranking on some prompts is treated as disproportionately costly. In either case, v_i formalizes the core auditing idea that the principal should spend effort where it matters for real-world outcomes, not where it is easiest to measure.

The principal's decision: allocating a finite label budget. The principal has a total labeling budget $K \in \mathbb{Z}_+$, interpreted as the total number of rater comparisons that can be purchased (due to cost, time, or governance-imposed limits). The principal chooses a nonnegative integer allocation $\{n_i\}_{i=1}^N$ satisfying

$$\sum_{i=1}^{N} n_i \;\leq\; K. \tag{5}$$

We allow the allocation to be adaptive: the principal may buy labels sequentially, updating posteriors after each comparison and deciding where to sample next based on observed outcomes. This adaptivity is important in practice—auditing is often exploratory—and it is also what makes "value of information" arguments bite: if we can stop sampling early on easy cases and redirect effort toward ambiguous ones, the same budget can yield substantially more decision-relevant certainty.

A minimal interface to training: from labels to post-training rankings. To connect auditing to behavior change, we need a reduced-form description of what the downstream preference-optimization step does with the collected labels. We take the downstream algorithm (DPO/RLHF with a fixed recipe) as given, and we care only about whether it induces the *correct ranking* on each audited comparison at deployment time. Accordingly, for each datapoint i we define an indicator event

$$R_i \;=\; \mathbf{1}\{\text{the trained policy ranks } y_i^w \text{ above } y_i^\ell \text{ given } x_i\}. \tag{6}$$

The principal's deployment welfare is then modeled additively as

$$W \;=\; \sum_{i=1}^{N} v_i R_i. \tag{7}$$

Additivity is a simplification—real systems exhibit cross-generalization and interference—but it is the right starting point for an *allocation* model: it lets us ask which datapoints have the highest marginal contribution to expected welfare under a fixed training recipe. In later sections we discuss how this abstraction can fail (e.g. when a small set of datapoints controls a global refusal boundary), and how to reinterpret i as indexing *clusters* or *features* rather than literal isolated prompt pairs.

What remains is to specify how R_i depends on α_i, the observed reference difficulty c_i, and the training recipe (notably the regularization/inertia parameter β). Our key modeling move is to assume that, conditional on the datapoint being trained on "enough," the algorithm has a deterministic success criterion of the form

$$R_i \;=\; \mathbf{1}\{\alpha_i > t_i\}, \tag{8}$$

where $t_i \in (0,1)$ is a learnability threshold that may depend on c_i and β. This captures the empirical idea from the previous section: under conservative, reference-conditioned updates, there is a boundary separating comparisons that the optimizer will reliably flip (given sufficient evidence) from those it will effectively preserve as in the reference. In Section 4 we derive, under an idealized DPO optimum, the specific threshold form $t_i = \sigma(\beta c_i)$ and interpret β as an adjustment cost that increases inertia around π_{ref}.

Auditing as posterior probability of learnability. Given the threshold template (8), the principal's uncertainty about post-training behavior on datapoint i reduces to posterior uncertainty about whether α_i exceeds t_i. Define the posterior tail probability

$$p_i(n_i) \;:=\; \Pr(\alpha_i > t_i \mid \mathcal{D}_i). \tag{9}$$

Then the principal's expected welfare contribution from datapoint i, after purchasing n_i labels and observing $\mathcal{D}_i$, is

$$\mathbb{E}[v_i R_i \mid \mathcal{D}_i] \;=\; v_i \, p_i(n_i). \tag{10}$$

This expression makes explicit why "more labels" is not uniformly valuable. A label matters when it appreciably changes $p_i(n_i)$, which occurs primarily when the posterior mass of α_i straddles the threshold t_i; when the datapoint is clearly learnable or clearly unlearnable under the training recipe, $p_i(n_i)$ is already near 1 or 0 and additional comparisons have diminishing returns.

In summary, the primitives $\{(c_i, v_i)\}$ are observable ex ante, the latent α_i governs rater outcomes via a Binomial model, and the principal allocates a finite budget K to form posteriors over α_i that determine expected deployment welfare through a (to-be-derived) learnability threshold. The next section provides the promised idealized link from reference-conditioning to the threshold t_i, which will let us turn this setup into a concrete indexable allocation problem.

4 Idealized learnability and the threshold form

We now make precise the "learnability threshold" t_i that appeared in the setup, specializing it to the case of an idealized DPO/RLHF update with reference-conditioning. The point of this section is not to claim that real training dynamics are literally separable across datapoints, but to extract a simple, decision-relevant statistic: given a comparison where the reference policy strongly prefers the *dispreferred* completion (large c_i), how much rater evidence is required before the training rule will reliably flip the model's ranking?

A single-datapoint reduction. Fix a datapoint (x, y^w, y^ℓ) and suppress the index i. Write the policy log-odds on this pair as

$$s_\theta \;:=\; \log \frac{\pi_\theta(y^w \mid x)}{\pi_\theta(y^\ell \mid x)}, \qquad s_{\text{ref}} \;:=\; \log \frac{\pi_{\text{ref}}(y^w \mid x)}{\pi_{\text{ref}}(y^\ell \mid x)} \;=\; -c. \tag{11}$$

A trained policy ranks y^w above y^ℓ if and only if $s_\theta > 0$. Our goal is to characterize the sign of s_θ under an idealized optimum of a reference-conditioned preference objective.

Following the analysis perspective in Chen et al. **?**, we treat DPO as fitting a logistic model to preference labels where the "feature" is the *change in log-odds relative to the reference*. Define the relative score

$$d_\theta \;:=\; s_\theta - s_{\text{ref}} \;=\; \log \frac{\pi_\theta(y^w \mid x)\, \pi_{\text{ref}}(y^\ell \mid x)}{\pi_\theta(y^\ell \mid x)\, \pi_{\text{ref}}(y^w \mid x)}. \tag{12}$$

In DPO, this quantity is passed through a logistic link with scale $\beta > 0$. Intuitively, d_θ measures how much the current policy has "tilted" toward the preferred completion relative to the baseline; β controls how aggressively such tilts are rewarded by the objective (or, in the RLHF view, how expensive deviations from π_{ref} are).

Expected DPO objective under Bradley–Terry labels. We assume the observed preference label is a Bernoulli random variable indicating whether

a rater prefers y^w to y^ℓ, with success probability $\alpha \in (0,1)$. Under the usual DPO log-likelihood form, the (per-datapoint) objective is proportional to

$$\ell(d_\theta) \;=\; \alpha \log \sigma(\beta d_\theta) \;+\; (1-\alpha) \log \sigma(-\beta d_\theta), \tag{13}$$

where $\sigma(z) = \frac{1}{1+e^{-z}}$. This is exactly the expected log-likelihood of a logistic classifier that predicts the preference outcome using βd_θ as its logit.[1]

The key observation is that $\ell(d_\theta)$ is strictly concave in d_θ and its maximizer is characterized by matching the model-implied win probability to the true win probability:

$$\sigma(\beta d_\theta^\star) \;=\; \alpha. \tag{14}$$

To see this, differentiate (13):

$$\frac{\partial \ell}{\partial d_\theta} \;=\; \beta\Big(\alpha - \sigma(\beta d_\theta)\Big), \tag{15}$$

so the unique stationary point satisfies (14).

Solving (14) yields an explicit optimal relative score:

$$d_\theta^\star \;=\; \frac{1}{\beta}\,\mathrm{logit}(\alpha) \;=\; \frac{1}{\beta}\log\frac{\alpha}{1-\alpha}. \tag{16}$$

Thus, under the idealized optimum, DPO shifts the policy's log-odds on this pair by an amount proportional to the rater log-odds $\mathrm{logit}(\alpha)$, with proportionality factor $1/\beta$.

From relative scores to a learnability threshold. Combining $s_\theta = s_{\mathrm{ref}} + d_\theta$ with (16) gives the idealized post-training log-odds

$$s_\theta^\star \;=\; s_{\mathrm{ref}} + \frac{1}{\beta}\mathrm{logit}(\alpha) \;=\; -c + \frac{1}{\beta}\mathrm{logit}(\alpha). \tag{17}$$

We obtain an immediate criterion for ranking correctness:

$$\begin{aligned} s_\theta^\star > 0 \;&\Longleftrightarrow\; -c + \frac{1}{\beta}\mathrm{logit}(\alpha) > 0 \\ &\Longleftrightarrow\; \mathrm{logit}(\alpha) > \beta c \\ &\Longleftrightarrow\; \alpha > \sigma(\beta c). \end{aligned} \tag{18}$$

This is the promised threshold form. In the notation of Section 3, we can read off $t = \sigma(\beta c)$ and so the idealized ranking event is $R = \mathbf{1}\{\alpha > t\}$. Importantly, the threshold depends on *observable* reference difficulty c and

[1] More precisely, one obtains (13) by taking the DPO objective and replacing empirical win rates with their population expectation α. This is the "infinite-data, no-generalization" idealization that lets us isolate the role of reference-conditioning.

the training recipe parameter β, while α is the latent property revealed only through labels.

Several sanity checks are worth noting. If $c = 0$ (the reference assigns equal probability to y^w and y^ℓ), then $t = \sigma(0) = 1/2$: a bare majority preference signal suffices. If $c > 0$ (the reference favors the dispreferred completion), then $t > 1/2$: we need *supermajority* agreement to overcome the reference. Conversely, if $c < 0$ (the reference already favors y^w), then $t < 1/2$: even modest rater evidence is enough to keep (or further reinforce) the correct ordering.

Interpretation: β as inertia and adjustment cost. Equation (17) makes the economic interpretation particularly transparent: the reference contributes a baseline log-odds margin $-c$, and the preference signal contributes an additive log-odds shift $\frac{1}{\beta}\mathrm{logit}(\alpha)$. Increasing β shrinks the magnitude of this shift for any fixed α; equivalently, it raises the threshold $t = \sigma(\beta c)$ whenever $c > 0$. In this sense, β is an *inertia* parameter: a larger β makes the optimizer more reluctant to move away from π_{ref} on comparisons where doing so would require flipping a reference-preferred ordering.

This matches the familiar RLHF interpretation in which one maximizes expected reward subject to a KL penalty to the reference. In the simplest two-action reduction (choose y^w versus y^ℓ at x), the KL-regularized optimum takes the exponential-tilting form

$$\pi^\star_\theta(\cdot \mid x) \;\propto\; \pi_{\mathrm{ref}}(\cdot \mid x) \exp\!\left(\frac{r(\cdot)}{\beta}\right), \tag{19}$$

so that the log-odds shift satisfies

$$s^\star_\theta - s_{\mathrm{ref}} \;=\; \frac{1}{\beta}\big(r(y^w) - r(y^\ell)\big). \tag{20}$$

If we take the Bradley–Terry preference rate α to be induced by an underlying reward gap via $\alpha = \sigma(r(y^w) - r(y^\ell))$, then $r(y^w) - r(y^\ell) = \mathrm{logit}(\alpha)$ and we recover (16) and (18). Under this view, β is literally the Lagrange multiplier trading off reward improvement against KL deviation, i.e. an adjustment cost measured in nats.

Safety implications and failure modes of the threshold picture. The threshold (18) formalizes a safety-relevant asymmetry: comparisons that the reference model already gets "mostly right" (negative or small c) are easy to preserve, while comparisons where the reference strongly prefers the dispreferred completion (large c) require very strong and reliable oversight signals to correct. This is desirable when high-c cases correspond to adversarial, ambiguous, or underspecified prompts where rater labels are noisy or exploitable: a conservative β prevents the policy from overreacting

to weak evidence. However, the same mechanism can entrench misalignment when the reference's high-c region corresponds to genuine systematic failures (e.g. rare but high-stakes safety issues, or out-of-distribution behaviors). In that regime, increasing β makes it *harder* to fix the very cases we most want to correct, unless we invest enough labeling to establish α well above the elevated threshold.

This exposes a governance-relevant tradeoff: labs often tune β to maintain linguistic quality and prevent reward hacking, but (18) suggests that doing so also determines which parts of the error surface are even *reachable* by preference optimization. When auditors report that "RLHF did not change behavior" on a class of failure cases, one plausible structural explanation is not merely insufficient data, but that those cases lie in a high-c region where $\sigma(\beta c)$ is close to one—so only near-unanimous preferences would move the trained policy.

Limitations of the idealization (and why we keep it anyway). The separable, per-datapoint optimum used above ignores the coupled nature of real training: parameters are shared across prompts, gradients interact, and the training set induces generalization rather than independent "flips." Moreover, the mapping from win rates to $\mathrm{logit}(\alpha)$ is only as good as the Bradley–Terry abstraction, and the DPO optimum may be unattainable with finite compute or imperfect optimization. Nonetheless, the threshold form serves as a compact interface between auditing and training. It turns the question "will more labels on i change deployment behavior?" into the probabilistic question "how likely is α_i to exceed $\sigma(\beta c_i)$?" This is precisely the quantity we will index in the next section when defining the audit value I_i and the marginal label value $\Delta_i(n_i)$.

5 An audit index from posterior learnability

Given the threshold form from Section 4, each datapoint i induces a binary (but latent) event,

$$L_i \;:=\; \mathbf{1}\{\alpha_i > t_i\}, \qquad t_i \;:=\; \sigma(\beta c_i),$$

where $L_i = 1$ means that, under the idealized update rule, the trained policy will rank y_i^w above y_i^ℓ on x_i. The principal never observes L_i directly; instead, they purchase noisy comparisons and maintain a posterior over α_i. This turns auditing into a posterior inference problem about whether α_i clears a *reference-conditioned* bar t_i that depends on the observable log-ratio c_i and the training hyperparameter β.

Closed-form posterior tail probability. Let $\mathcal{D}_i = (w_i, n_i)$ be the observed label data for datapoint i, and define posterior parameters

$$a_i \;:=\; a_0 + w_i, \qquad b_i \;:=\; b_0 + n_i - w_i, \qquad \alpha_i \mid \mathcal{D}_i \sim \mathrm{Beta}(a_i, b_i).$$

The core statistic we will use is the posterior probability that i is learnable:

$$p_i(n_i) \;:=\; \Pr(\alpha_i > t_i \mid \mathcal{D}_i) \;=\; \int_{t_i}^{1} \frac{u^{a_i-1}(1-u)^{b_i-1}}{B(a_i, b_i)}\, du \;=\; 1 - I_{t_i}(a_i, b_i), \tag{21}$$

where $B(\cdot,\cdot)$ is the Beta function and $I_t(a,b)$ is the regularized incomplete beta function (i.e. the Beta CDF). Equation (21) is useful operationally: it is a one-line computation (available in standard numerical libraries), and it cleanly separates (i) reference difficulty via $t_i = \sigma(\beta c_i)$ from (ii) evidence strength via (a_i, b_i).

Two immediate comparative statics match the informal deployment story. First, holding $\mathcal{D}_i$ fixed, increasing c_i (or β) increases t_i and weakly decreases $p_i(n_i)$: if the reference model strongly prefers y_i^ℓ, we require stronger rater agreement to believe training will flip the ranking. Second, holding t_i fixed, increasing n_i concentrates the posterior and pushes $p_i(n_i)$ toward either 0 or 1 depending on whether the realized win rate lies below or above the threshold.

From posterior learnability to an audit index. The quantity $v_i p_i(n_i)$ is the expected welfare contribution of datapoint i under the idealized pipeline *if we stop labeling now*. However, for auditing we typically care about *where additional labels are decision-relevant*. Intuitively, datapoints fall into three regimes:

1. *Confidently learnable:* the posterior puts almost all mass above t_i ($p_i \approx 1$).
2. *Confidently unlearnable:* the posterior puts almost all mass below t_i ($p_i \approx 0$).
3. *Contested:* substantial posterior mass lies on both sides of t_i ($p_i \approx 1/2$).

Only the contested regime is one where an audit can plausibly change our conclusion about whether this datapoint is reachable by preference optimization. This motivates an *audit index* that upweights posterior disagreement about L_i. A canonical choice is the posterior variance of the learnability indicator, scaled by welfare:

$$I_i \;:=\; v_i \,\mathrm{Var}(L_i \mid \mathcal{D}_i) \;=\; v_i\, p_i(n_i)\big(1 - p_i(n_i)\big). \tag{22}$$

This index is maximized at $p_i = 1/2$ and vanishes as $p_i \to 0$ or 1. In other words, I_i prioritizes datapoints where (i) the deployment weight v_i is large and (ii) we are currently *uncertain* whether training will correct the ordering.

While (22) is not the only sensible index, it captures a safety-relevant notion of "known unknowns": if a high-stakes behavior is either clearly fixable or clearly not fixable (under the idealization), then extra auditing has low informational value; if it is ambiguous relative to the threshold, auditing is valuable because it can resolve whether the pipeline will actually move the deployed model in the desired direction.

How disagreement enters through mass near the threshold. The reason $p_i(1-p_i)$ is the right qualitative shape can be seen by looking at how sensitive p_i is to perturbations in either the threshold t_i or the posterior location. Differentiating (21) with respect to t_i yields

$$\frac{\partial p_i}{\partial t_i} \;=\; -f_{\mathrm{Beta}}(t_i; a_i, b_i), \tag{23}$$

where $f_{\mathrm{Beta}}(\cdot; a_i, b_i)$ is the Beta density. Thus, whenever the posterior density is large at the threshold (i.e. substantial probability mass sits *near* t_i), small changes to the threshold or to posterior parameters can materially change p_i. This is exactly the "disagreement" condition: the posterior neither safely clears the bar nor safely misses it; it is concentrated *around* the bar.

A complementary view uses a normal approximation for moderate n_i. If $\alpha_i \mid \mathcal{D}_i \approx \mathcal{N}(m_i, s_i^2)$ with mean $m_i = \frac{a_i}{a_i+b_i}$ and variance $s_i^2 \approx \frac{m_i(1-m_i)}{a_i+b_i+1}$, then

$$p_i(n_i) \;\approx\; 1 - \Phi\!\left(\frac{t_i - m_i}{s_i}\right), \qquad \frac{\partial p_i}{\partial m_i} \;\approx\; \frac{1}{s_i}\,\phi\!\left(\frac{t_i - m_i}{s_i}\right),$$

so the sensitivity is governed by the Gaussian pdf ϕ evaluated at the standardized gap $(t_i - m_i)/s_i$. Again, the expected impact of additional evidence is largest when the threshold lies within roughly one posterior standard deviation of the posterior mean, i.e. when the posterior places significant mass near t_i.

Marginal value of one additional label. To move from a static audit index to a decision rule for buying labels, we define the expected marginal gain from purchasing one more comparison on i. Under the Beta–Bernoulli model, the posterior predictive probability that the next rater prefers y_i^w is the posterior mean

$$\mu_i \;:=\; \mathbb{E}[\alpha_i \mid \mathcal{D}_i] \;=\; \frac{a_i}{a_i + b_i}.$$

If the next label is a win, the posterior becomes $\mathrm{Beta}(a_i + 1, b_i)$; if it is a loss, it becomes $\mathrm{Beta}(a_i, b_i + 1)$. Define the corresponding updated tail probabilities

$$p_i^+ \;:=\; \Pr(\alpha_i > t_i \mid a_i{+}1, b_i) \;=\; 1{-}I_{t_i}(a_i{+}1, b_i), \qquad p_i^- \;:=\; \Pr(\alpha_i > t_i \mid a_i, b_i{+}1) \;=\; 1{-}I_{t_i}(a_i, b_i{+}1).$$

Then the (one-step) marginal value of an extra label is

$$\Delta_i(n_i) \;:=\; \mathbb{E}[v_i p_i(n_i+1) - v_i p_i(n_i) \mid \mathcal{D}_i] \;=\; v_i\Big(\mu_i p_i^{+} + (1-\mu_i)p_i^{-} - p_i\Big). \tag{24}$$

Equation (24) makes two features explicit. First, v_i scales value linearly: high-frequency or high-stakes datapoints are always more attractive to audit, holding everything else fixed. Second, $\Delta_i(n_i)$ is largest precisely when an additional Bernoulli observation is likely to move substantial posterior mass across the threshold. In the extreme regimes, $\Delta_i(n_i)$ is small: if $p_i \approx 1$, then both p_i^{+} and p_i^{-} are close to 1; if $p_i \approx 0$, both updates are close to 0. The only way for the expectation in (24) to be sizeable is for the posterior to be "poised" near t_i, so that a single win versus loss meaningfully changes $\Pr(\alpha_i > t_i)$.

This connects back to reference difficulty c_i. Since $t_i = \sigma(\beta c_i)$, large c_i pushes t_i toward 1, meaning that only datapoints with strong evidence of near-unanimous rater agreement will yield p_i near one. In practice, this creates a characteristic auditing pattern: high-c_i regions tend to be either confidently unlearnable (if raters do not strongly agree) or highly contested (if evidence is limited but plausible), and (24) prioritizes the latter. This is the sense in which optimal auditing (in our model) oversamples high-c_i and high-disagreement regions relative to uniform sampling.

Safety and governance interpretation. The pair (p_i, Δ_i) supports a clean separation between *status* and *leverage*. The tail probability p_i answers: "How likely is it that our current training recipe can fix this behavior?" The marginal value Δ_i answers: "Is it worth spending one more comparison to refine that answer?" This is useful for governance because it provides an auditable, quantitative rationale for label allocation that is not merely "collect more data" but "collect data where the training pipeline is near an actionable boundary." At the same time, this picture inherits all of the usual failure modes: if rater noise is systematically biased, if α_i is not well-modeled by i.i.d. Bradley–Terry labels, or if the idealized threshold rule mispredicts real training dynamics, then p_i and Δ_i can be miscalibrated. Those issues do not eliminate the value of an index; they instead shift the burden to robustness (e.g. hierarchical priors, rater modeling, adversarial evaluation) and to validating that the threshold approximation remains predictive in the regimes we intend to audit.

In the next section, we treat $\{\Delta_i(n_i)\}$ as the primitives of a label allocation problem: we contrast a one-shot knapsack view with a sequential (bandit-style) view, and we characterize when greedy index policies are provably near-optimal under monotonicity and diminishing-returns conditions.

6 Label allocation: knapsack versus sequential (bandit) auditing

We now formalize the principal's label-allocation problem. The key object from the previous section is the posterior tail probability $p_i(n_i) = \Pr(\alpha_i > t_i \mid \mathcal{D}_i)$, which converts noisy comparisons into a belief about whether datapoint i clears the *reference-conditioned* learnability threshold $t_i = \sigma(\beta c_i)$. Given welfare weights $\{v_i\}$, this induces an additive welfare surrogate $\sum_i v_i p_i(n_i)$, and the central operational question becomes: *how should we spend a finite budget of comparisons K across datapoints to maximize expected downstream welfare under this surrogate?*

Two allocation models: one-shot (knapsack) versus sequential (adaptive). There are two natural ways to pose the decision problem, corresponding to different operational pipelines.

(i) One-shot allocation (knapsack view). The principal commits ex ante to an integer allocation $n = (n_1, \ldots, n_N) \in \mathbb{Z}_+^N$ satisfying $\sum_i n_i \le K$. Since outcomes are stochastic, the resulting posteriors (and thus $\{p_i(n_i)\}$) are random, so the ex ante objective is

$$\max_{n \in \mathbb{Z}_+^N : \sum_i n_i \le K} \quad \sum_{i=1}^{N} \mathbb{E}[v_i \, p_i(n_i)] \,, \tag{25}$$

where the expectation is over the label draws induced by the (unknown) α_i under the assumed data-generating process. This is a knapsack-style resource allocation problem with *separable* value functions.

(ii) Sequential allocation (bandit view). The principal chooses comparisons one at a time. At step $k \in \{1, \ldots, K\}$, they select an index i_k, observe the win/loss outcome, update (a_i, b_i), and proceed. The resulting policy π maps the current collection of posteriors (equivalently, the vector of Beta parameters $\{(a_i, b_i)\}$) to a choice of which datapoint to label next. The objective becomes the expected terminal welfare surrogate

$$\max_{\pi} \quad \mathbb{E}_\pi \left[\sum_{i=1}^{N} v_i \, p_i(n_i(K)) \right] , \tag{26}$$

where $n_i(K)$ denotes the (random) number of labels allocated to datapoint i by time K. This is a finite-horizon Bayesian bandit/MDP: each datapoint is an "arm" with a posterior state (a_i, b_i), and sampling an arm produces a Bernoulli outcome that updates only that arm.

The one-shot formulation is appropriate when procurement requires committing to batches (e.g. contracting for n_i labels per cluster). The sequential formulation is appropriate when we can route tasks online and react to early evidence (e.g. stopping early on clearly learnable/unlearnable points and reallocating to contested ones).

A separable concave resource allocation problem (and why concavity matters). Both (25) and (26) become tractable—and admit index-like solutions—when the per-datapoint value of additional labels exhibits *diminishing returns*. To express this in the one-shot setting, define

$$g_i(n) \;:=\; \mathbb{E}[v_i\, p_i(n)]\,, \qquad n \in \mathbb{Z}_+,$$

and its discrete marginal increments

$$\bar{\Delta}_i(n) \;:=\; g_i(n+1) - g_i(n) \;=\; \mathbb{E}[v_i\, p_i(n+1) - v_i\, p_i(n)]\,. \tag{27}$$

A natural regularity condition is discrete concavity of g_i, equivalently $\bar{\Delta}_i(n)$ nonincreasing in n. Intuitively, once we have purchased many comparisons on i, the posterior about whether α_i clears t_i becomes tight, so the informational (and thus welfare) benefit of one additional label should shrink.

This diminishing-returns condition is also what distinguishes the present problem from a generic knapsack. With arbitrary $g_i(\cdot)$, (25) can encode hard combinatorial structure. With separability plus concavity, the problem becomes a form of *discrete concave resource allocation*, for which simple greedy rules and exchange arguments often apply (and, in the adaptive case, motivate index policies).

Continuous relaxation and KKT characterization (shadow price of labels). A standard way to expose the structure is to relax the integrality constraint and allow $n_i \in \mathbb{R}_+$. Writing the relaxed objective as $\max_{n \geq 0,\, \sum n_i \leq K} \sum_i g_i(n_i)$, assume g_i is concave and differentiable on $\mathbb{R}_+$ (e.g. justified by a normal approximation to the Beta posterior for moderate n). The Lagrangian is

$$\mathcal{L}(n, \lambda) \;=\; \sum_{i=1}^{N} g_i(n_i) - \lambda\Big(\sum_{i=1}^{N} n_i - K\Big), \qquad \lambda \geq 0.$$

The KKT conditions imply that at an optimal relaxed solution n^* there exists λ^* such that, for each i,

$$g_i'(n_i^*) \;\leq\; \lambda^*, \qquad \text{with equality if } n_i^* > 0, \tag{28}$$

and $\sum_i n_i^* = K$ whenever $\lambda^* > 0$. Economically, λ^* is the *shadow price* of one additional comparison: we allocate labels to datapoints until their marginal welfare gain equals the common price λ^*. Datapoints whose marginal value is everywhere below λ^* receive zero labels; datapoints with high v_i and posterior mass near the threshold t_i tend to have high marginal value and thus receive more.

In the integer problem, an analogous characterization holds using discrete marginals: for an "active" i, the last allocated label is one whose $\bar{\Delta}_i(n_i - 1)$ is (approximately) at the cutoff, and the next label $\bar{\Delta}_i(n_i)$ is (approximately) below it. This is the discrete counterpart of (28) and motivates incremental label allocation rules.

Greedy/index policies as incremental solutions. The KKT picture suggests an incremental implementation: allocate labels one at a time to the datapoint with the largest current marginal value.

In a *one-shot* (non-adaptive) implementation, one would conceptually rank all unit-increments (i, n) by $\bar{\Delta}_i(n)$ and take the top K. When each g_i is concave, this incremental greedy procedure is closely related to classical algorithms for separable concave allocation: adding one unit at a time to the coordinate with the largest remaining marginal automatically equalizes marginals, mirroring (28).

In a *sequential* (adaptive) implementation, we instead use the posterior-conditioned one-step marginal defined in (24):

$$\Delta_i(n_i) \;=\; \mathbb{E}[v_i p_i(n_i + 1) - v_i p_i(n_i) \mid \mathcal{D}_i]\,,$$

which is computable from the current Beta parameters (a_i, b_i) via the two updated tail probabilities p_i^+, p_i^-. An index policy then takes the form

$$i_k \in \arg\max_{i \in \{1,\ldots,N\}} \Delta_i(n_i(k-1)), \tag{29}$$

with ties broken arbitrarily (or by secondary criteria such as preferring larger v_i or larger c_i). Operationally, (29) is attractive because it is *local*: it requires only maintaining per-datapoint posteriors and recomputing Δ_i for the chosen i after each label.

This greedy rule is also the cleanest embodiment of the "audit where labels are decision-relevant" principle. If the posterior is already far above or far below the threshold t_i, then (typically) $p_i^+ \approx p_i^- \approx p_i$, so Δ_i is small. Conversely, if the posterior places substantial mass near t_i, then p_i^+ and p_i^- can differ materially, and Δ_i is large: one more label has real leverage to move our belief about learnability.

When is greedy sensible? Monotonicity and diminishing returns. Greedy and index policies rely on two qualitative properties.

Monotonicity. We want the value of additional labels to be nonnegative in expectation: $g_i(n + 1) \geq g_i(n)$, or at least that negative increments are rare and dominated by positive ones when averaged over uncertainty. In deployment terms, this says that collecting more evidence should not systematically reduce expected welfare—it should either increase it (by revealing learnability we can exploit) or leave it roughly unchanged (by confirming what we already knew).

Diminishing returns. We want $\bar{\Delta}_i(n)$ (and often its posterior-conditioned analog $\Delta_i(n)$) to decrease as n grows. In the Beta–Bernoulli model, posterior concentration increases like $1/(a_i + b_i)$, so the probability mass *near* the threshold t_i typically shrinks with n; this is the informal driver of diminishing returns. Under a normal approximation $\alpha_i \mid \mathcal{D}_i \approx \mathcal{N}(m_i, s_i^2)$, the

"actionable" region is where $|t_i - m_i| \lesssim s_i$, and s_i decays on the order of $1/\sqrt{n_i}$, shrinking the region in which another label can change conclusions.

These conditions are not purely mathematical conveniences; they are also safety-relevant. If diminishing returns fails (e.g. due to distribution shift across raters, nonstationary α_i, or systematic annotation artifacts), then an index policy can get stuck oversampling misleading regions or chasing spurious uncertainty. In such cases, it can be necessary to augment the model (hierarchical priors, rater mixture models, drift detection) or to impose governance constraints (minimum auditing across critical clusters) that break the purely myopic logic.

Algorithmic spectrum: from knapsack solvers to Bayesian bandits. Finally, it is useful to situate greedy within a spectrum of solution concepts. At one end, if we treat (25) as a deterministic knapsack with precomputed values $g_i(n)$, then dynamic programming or convex-cost flow methods can be used when N and K are modest. At the other end, the fully adaptive problem (26) is, in principle, solvable by dynamic programming on the full posterior state, but the state space grows quickly with N and K. Index policies like (29) occupy the pragmatic middle: they are computationally light, easy to audit, and often near-optimal when monotonicity and diminishing returns hold.

In the next section we make this precise: we state sufficient conditions (submodularity/concavity) under which greedy achieves formal approximation guarantees, and we discuss when those conditions break—necessitating heavier numerical machinery (e.g. bandit relaxations, lookahead, or problem-specific dynamic programs).

7. Near-optimality of greedy: concavity, (adaptive) submodularity, and when we need heavier machinery. The previous section motivated the incremental rule (29) on economic grounds (equalizing marginal value across datapoints). We now make the corresponding performance claim more explicit. The core message is that *diminishing returns* is not just a heuristic: under standard concavity/submodularity conditions it yields formal approximation guarantees for greedy, while violations of these conditions are precisely the regimes where we should expect to need dynamic programming, bandit relaxations, or explicit lookahead.

Offline (one-shot) view: separable concave allocation is essentially "already solved". Consider first the relaxed one-shot objective (25) with separable values $g_i(n) = \mathbb{E}[v_i p_i(n)]$. When each g_i is concave (in the discrete sense that $\bar{\Delta}_i(n) = g_i(n+1) - g_i(n)$ is nonincreasing), the resulting integer allocation problem is a classical *separable concave resource allocation* problem. In this setting, the unit-cost greedy algorithm that repeatedly assigns

the next label to the coordinate with largest remaining marginal $\bar{\Delta}_i(n_i)$ is not merely a $(1 - 1gP3)$-approximation; it is (under mild tie-breaking conditions) *optimal* for the integer problem. Intuitively, concavity rules out "complementarities" across labels on the same datapoint: if the first few labels on i are valuable, then later labels on i cannot become *more* valuable than earlier ones. This exchange property is exactly what greedy exploits.

Formally, if we represent an allocation by the multiset of K unit increments $\{(i, 1), \ldots, (i, n_i)\}$ for each i, concavity ensures that the sorted list of all marginals $\{\bar{\Delta}_i(n)\}_{i,n}$ is "prefix optimal": taking the top K marginals yields an allocation that cannot be improved by swapping any chosen unit with any unchosen unit. This is the discrete analog of the KKT equal-marginal condition (28) and explains why, in the one-shot model, greedy is a principled baseline rather than an ad hoc heuristic.

Set-function view: monotone submodularity yields the $(1 - 1/e)$ guarantee. The preceding argument leverages separability. To understand guarantees that survive beyond strict separability (or that extend to richer objectives), it is useful to re-encode the allocation problem as maximization of a set function. Let $\mathcal{E}$ denote the ground set of available "labeling actions" and let $S \subseteq \mathcal{E}$ denote the set of K actions we take. Define $F(S)$ as the expected terminal welfare surrogate after executing S (with the understanding that S may contain multiple labels for the same datapoint, so technically S can be treated as a multiset, or one can expand $\mathcal{E}$ into K time-stamped copies). In many information-gathering problems, F is *monotone submodular*: (i) monotonicity says additional labels do not reduce expected value, $F(S \cup \{e\}) \geq F(S)$; (ii) submodularity says marginal gains diminish with more information, $F(S \cup \{e\}) - F(S) \geq F(T \cup \{e\}) - F(T)$ whenever $S \subseteq T$.

When F is monotone submodular and we have a cardinality constraint $|S| \leq K$, the standard Nemhauser–Wolsey result implies that the greedy algorithm achieves

$$F(S_{\text{greedy}}) \;\geq\; (1 - 1/e)\, F(S^\star),$$

where $S^\star$ is an optimal set of size K. In our context, this guarantee can be read as a robustness statement: even if the per-datapoint value is not perfectly concave or perfectly separable, as long as the overall value of information exhibits diminishing returns in the submodular sense, greedy remains provably near-optimal.

It is important, however, to be honest about what must be true for monotone submodularity to hold. The functional $p_i(n) = \Pr(\alpha_i > t_i \mid \mathcal{D}_i)$ is a *thresholded* posterior quantity, and thresholded objectives can create regions where additional data has highly nonlinear effects (e.g., when the posterior mean is near t_i). Submodularity is therefore not automatic; it is best viewed as an approximation justified by posterior concentration: once n

is moderate, additional labels mostly *sharpen* an already-unimodal posterior, making the marginal value of data naturally decrease.

Sequential (adaptive) view: adaptive submodularity explains why myopic sampling works. The sequential formulation (26) is more delicate because the principal conditions on realized outcomes. In this setting, the appropriate concept is *adaptive submodularity* (Golovin–Krause): the expected marginal benefit of selecting an action e should decrease as we condition on a *richer* history. Concretely, let ψ denote the current posterior state (here, the collection of Beta parameters $\{(a_i, b_i)\}$). Define the conditional marginal value of sampling datapoint i once more as the posterior one-step gain $\Delta_i(n_i)$ already used in (29). Adaptive submodularity asserts that, as ψ becomes more informative (more labels observed), the conditional expected gain from an additional sample does not increase.

Under adaptive monotonicity and adaptive submodularity, the adaptive greedy policy (29) enjoys the same $(1-1/e)$ approximation guarantee relative to the optimal adaptive policy (up to technicalities about ties and feasibility). The interpretation is exactly the one we want for auditing: *if* labels have diminishing decision relevance as evidence accumulates, then it is safe (in the approximation sense) to spend budget myopically on whichever datapoint is currently most likely to flip our "learnable vs. not" conclusion.

Why these sufficient conditions are safety-relevant. The concavity/submodularity assumptions are not merely mathematical conveniences; they encode a substantive claim about how evidence interacts with downstream training: that information value saturates rather than exhibiting strong complementarities or delayed payoffs. From a safety perspective, the danger is that if the assumptions fail, greedy can become systematically miscalibrated in ways that matter for oversight.

Three failure modes are worth flagging.

1. *Non-monotonic value of evidence.* If additional labels can reduce expected welfare (for example because they reveal that a datapoint is *unlearnable* under the current β and reference model, or because the training pipeline reacts pathologically to certain labeled examples), then the objective is not monotone. Greedy policies can still be used, but classical guarantees do not apply and one should expect regimes where "stop labeling" or "label elsewhere" dominates.

2. *Complementarities across datapoints (non-separability).* Our surrogate $\sum_i v_i p_i(n_i)$ assumes independent contributions. In reality, gradient updates from one region of the data distribution can generalize (positively or negatively) to another. Such cross-effects create complementarities

(or interference) across datapoints that violate submodularity. Oversight implication: we may need to audit *clusters* jointly, or explicitly model transfer, rather than treat datapoints as independent arms.

3. *Nonstationarity and rater mixture structure.* If α_i drifts over time (policy-dependent preferences, changing rater populations) or labels are generated by heterogeneous subpopulations, then the Beta–Bernoulli conjugate model is misspecified. Misspecification can create spurious "persistent uncertainty" where additional labels keep changing conclusions, breaking diminishing returns and potentially causing greedy to chase noise.

When we should expect to need numerical methods (DP, bandits, lookahead). When the sufficient conditions do not hold, the principal's problem becomes closer to a general finite-horizon Bayesian bandit/MDP. Exact solutions require dynamic programming over a state space that scales with $\prod_i (n_i + 1)$, which is quickly infeasible as N and K grow. In these regimes, we see three pragmatic escalations.

(i) Small-scale exact or near-exact DP. When N is small (e.g. auditing a small number of high-stakes clusters) and K is modest, value iteration or backward induction on the Beta parameters is feasible. This can serve as a gold standard to evaluate greedy/index policies and to quantify the cost of myopia.

(ii) Bandit relaxations and indices. For longer horizons, one can consider discounted or infinite-horizon relaxations where indexability emerges (e.g. Gittins-style indices). While our objective is terminal rather than discounted, such relaxations can provide computable heuristics with some theoretical backing. A related approach is Lagrangian relaxation: treat the budget constraint via a penalty λ and solve decoupled per-arm problems to derive a Whittle-like index, then tune λ to meet the budget.

(iii) Approximate planning. When the environment is misspecified or strongly non-submodular, Monte Carlo tree search, limited lookahead, or rollout policies (greedy plus one-step or few-step planning) can materially improve performance. These methods are also compatible with governance constraints (e.g. minimum auditing quotas per safety-critical slice), which can be integrated as hard constraints in the planner rather than as after-the-fact patches.

Taken together, the near-optimality results give us a disciplined story: greedy is not a leap of faith but a consequence of diminishing returns in the value of evidence. At the same time, the precise ways in which the assumptions fail point directly to the operational situations where we should invest in richer modeling, stronger oversight constraints, or heavier numerical optimization.

8. Comparative statics and design implications: how design choices tilt the audit index. The index rule is only as useful as our understanding of how it changes when we touch the knobs the training pipeline actually exposes. In our reduced model those knobs appear in three places: (i) the regularization/inertia parameter β, which converts reference log-ratios c_i into a learnability threshold $t_i = \sigma(\beta c_i)$; (ii) the quality and calibration of the reference policy π_{ref}, which determines the *distribution* and *scale* of the observable c_i; and (iii) the labeling protocol, which governs the effective noise in rater outcomes and hence the posterior concentration of α_i as n_i grows. Comparative statics along these axes tell us, operationally, when an "audit-the-disagreement" strategy is sufficient, and when we should instead treat auditing as a targeted effort to overcome reference inertia on particular slices.

How β reshapes the threshold and reorders what is worth auditing. Because $t_i = \sigma(\beta c_i)$, the mechanical sensitivity is

$$\frac{\partial t_i}{\partial \beta} \;=\; c_i\,\sigma(\beta c_i)\big(1-\sigma(\beta c_i)\big) \;=\; c_i\,t_i(1-t_i), \qquad \frac{\partial t_i}{\partial c_i} \;=\; \beta\,t_i(1-t_i).$$

Two qualitative consequences follow immediately. First, increasing β polarizes the threshold: for $c_i > 0$ (the reference prefers y_i^ℓ), the threshold rises toward 1, making such datapoints harder to "flip" via preference training; for $c_i < 0$, the threshold falls toward 0, making such points essentially automatically learnable. Second, the sensitivity is largest when $\beta c_i \approx 0$, i.e. precisely when the reference is uncertain (or indifferent) between y_i^w and y_i^ℓ. Thus β does not just scale difficulty uniformly; it reallocates difficulty mass from the "near-tie" region to the "reference-confident-but-wrong" region.

This feeds directly into the audit index through $p_i(n) = \Pr(\alpha_i > t_i \mid \mathcal{D}_i)$. A useful approximation for intuition is a normal approximation to the Beta posterior: if $\alpha_i \mid \mathcal{D}_i \approx \mathcal{N}(\mu_i, \tau_i^2)$ (with τ_i^2 shrinking like $1/n_i$), then

$$p_i(n_i) \;\approx\; 1 - \Phi\left(\frac{t_i - \mu_i}{\tau_i}\right),$$

so the marginal value of additional labels is largest when $(t_i - \mu_i)/\tau_i$ is near 0, i.e. when the posterior mass straddles the threshold. Increasing β moves t_i upward for $c_i > 0$, meaning that (holding μ_i, τ_i fixed) the posterior must be pushed further upward—requiring more evidence—to reach the same tail probability. In welfare terms, a larger β shifts the optimal allocation toward (a) higher v_i and (b) larger positive c_i points where μ_i is plausibly high but we need more labels to prove $\alpha_i > t_i$. In contrast, when β is small, $t_i \approx 1/2$ across a wide range of c_i, so auditing behaves more like generic "is the preference direction stable?" sampling, with less emphasis on overriding the reference.

A design implication is that β and K are coupled choices. If governance or product constraints force a large β (high inertia to preserve reference behavior), then auditing must budget for the fact that high-c overrides become label-expensive; if we cannot afford that, then the implied deployment behavior is that the system will systematically *refuse to learn* from human preferences in precisely the regimes where the reference is most in conflict with them. Conversely, if we lower β to make more datapoints learnable at fixed K, we should expect stronger sensitivity to residual rater noise and distribution shift, and we may need additional safeguards (held-out evaluations, conservative updates, or explicit constraints) to avoid overcorrecting on thin evidence.

Reference strength and calibration: why "better π_{ref}" is not one-dimensional. The observable $c_i = \log \frac{\pi_{\text{ref}}(y_i^\ell|x_i)}{\pi_{\text{ref}}(y_i^w|x_i)}$ plays two roles: it encodes which side the reference favors and how confidently. If π_{ref} improves in the usual sense (higher likelihood on genuinely preferred completions), then the mass of datapoints with $c_i > 0$ should shrink, and many remaining c_i will become more negative. Under our thresholding, this makes more of the distribution "trivially learnable" (since t_i becomes small), which *reduces* the need for auditing those regions: the pipeline will get them right even with weak preference evidence.

However, there is an important countervailing effect: reference *confidence* interacts with β. A reference that is occasionally wrong but very confident when wrong produces large positive c_i outliers. Because $t_i = \sigma(\beta c_i)$ is steep in βc_i, these outliers can become practically unlearnable unless α_i is extremely close to 1. In oversight terms, improving the reference without controlling calibration can concentrate residual risk into a thinner but higher-stakes tail: fewer failures, but failures that are harder for preference optimization to repair. This is exactly the regime where an index that upweights high c_i is most valuable, because uniform sampling will miss the tail.

Operationally, this suggests treating reference *calibration* as a first-class audit target. If we can reduce the magnitude of erroneous positive c_i (e.g. via temperature scaling, better uncertainty estimation, or a reference ensemble that dampens overconfidence), then for fixed β we lower t_i on the problematic cases and make them label-feasible. Said differently: at fixed label budget, there is a three-way trade among (i) reference calibration, (ii) β, and (iii) achievable coverage of "override" slices.

Rater noise and protocol choice: labels as information vs. labels as decisions. In our baseline model, each label is Bernoulli(α_i), so "noise" is subsumed into α_i itself (values near 1/2 correspond to high disagreement or ambiguity). In practice we often have additional, avoidable noise: in-

consistent instructions, low-effort ratings, presentation effects, or systematic subgroup differences. A convenient way to separate these is to posit a latent "clean" preference probability $\tilde{\alpha}_i$ and a protocol-dependent flipping rate $\eta \in [0, 1/2)$, so that the observed label probability becomes

$$\alpha_i \;=\; (1-\eta)\tilde{\alpha}_i + \eta(1-\tilde{\alpha}_i) \;=\; \eta + (1-2\eta)\tilde{\alpha}_i.$$

Under this model, improving the protocol (reducing η) stretches α_i away from $1/2$, which has two benefits for auditing. First, it increases the chance that α_i clears the threshold t_i on genuinely learnable points (especially important when $t_i > 1/2$ due to $c_i > 0$). Second, it increases per-label information: posteriors concentrate faster around the true α_i, raising $\Delta_i(n_i)$ early in the process and making greedy allocation more decisive.

This motivates a cost-quality trade: rather than spending budget K on many low-quality labels, the principal may prefer fewer higher-quality comparisons (expert raters, adjudication, better task design) if the effective reduction in η is large. In the index language, changing the protocol changes the entire curve $n \mapsto \mathbb{E}[p_i(n)]$, not just the realized w_i. A practical governance rule is therefore: treat protocol upgrades as multiplicative on *all* high-c, high-v slices, because those are exactly where the required posterior evidence to establish $\alpha_i > t_i$ is most demanding.

Joint design of β and auditing: "how much inertia can we afford?" Because the learnability condition is $\alpha_i > \sigma(\beta c_i)$, for any fixed (α_i, c_i) with $c_i > 0$ there is an implicit upper bound on β:

$$\beta \;<\; \frac{1}{c_i} \log \frac{\alpha_i}{1-\alpha_i}.$$

While α_i is unknown, the posterior gives a distribution over admissible β values for each datapoint. This suggests a design workflow that is more explicit than "pick β and hope": choose β to keep the set of high-welfare points plausibly learnable under the posterior, then allocate labels to resolve the remaining uncertainty. Concretely, if we define a target coverage constraint such as $\sum_i v_i \Pr(\alpha_i > t_i \mid \mathcal{D}_i) \geq \tau$, then β becomes a policy lever for trading off reliance on π_{ref} against the attainable audited improvement. This makes the safety trade-off legible: larger β protects against spurious overrides but risks baking in reference failures; smaller β enables correction but demands stronger controls on rater quality and distributional robustness.

Iterative and on-policy refresh: when c_i should be recomputed, and what that does to oversight. So far c_i is defined with respect to a fixed π_{ref}, but in many pipelines the "reference" is periodically refreshed (e.g. set to a recent checkpoint) to keep KL costs meaningful and to avoid training pathologies. Under refresh, both the c_i values and the underlying α_i can

drift (since the model generates different candidates and raters may respond differently to different outputs). Our index logic still applies, but the object being optimized becomes time-dependent: we are no longer allocating labels to estimate a static α_i relative to a static threshold; we are allocating labels to track a moving decision boundary.

Two implications matter for protocol design. First, refresh tends to shrink the magnitude of c_i on-policy (the current model is closer to itself than to an old anchor), which pushes t_i back toward $1/2$ and makes more points "contestable" by labels. This can improve sample efficiency, but it also means auditing becomes less targeted toward correcting old reference mistakes and more about detecting local preference gradients. Second, refresh can create a governance hazard: if π_{ref} is allowed to move freely, then failures that were once high-c (clearly in conflict with the anchor reference) may disappear from the audit surface, even though the deployed policy still exhibits problematic behavior relative to human preferences. A conservative compromise is to maintain two references: a moving $\pi_{\text{ref}}^{\text{train}}$ for optimization stability and an anchored $\pi_{\text{ref}}^{\text{audit}}$ for longitudinal accountability, and to compute complementary indices against each.

What these statics give us going into evaluation. The overarching lesson is that the audit index is not a fixed recipe; it is a diagnostic for where the pipeline is *structurally* unable to learn at a given $(\pi_{\text{ref}}, \beta, \text{protocol}, K)$. By varying β, perturbing reference quality (including calibration), and modeling rater noise, we can predict when optimal auditing should concentrate on high-c tail slices versus broadly sampling around ambiguity. The next step is to test whether these predicted reallocations translate into measurable gains on real preference corpora and realistic disagreement patterns, and to quantify how sensitive the gains are to reference quality and the choice of β.

9. Empirical evaluation plan: simulating audit-driven acquisition on real preference corpora. Our theory makes a sharp prediction about where labels are most valuable: not where preferences are merely noisy, but where (i) deployment welfare weight is high, (ii) the reference is confident in the wrong direction (large positive c_i), and (iii) the posterior mass over α_i still straddles the learnability threshold $t_i = \sigma(\beta c_i)$. The most direct way to test this is an *offline* simulation on existing multi-annotated preference datasets, where we can treat cross-annotations as ground truth about α_i and then "replay" alternative acquisition policies under a fixed label budget K. We complement this with a smaller number of *end-to-end* preference-optimization runs to check that the index gains survive the idealization.

Datasets and basic construction of (x_i, y_i^w, y_i^ℓ). We will use corpora that include repeated comparisons for the same prompt-output pair, such as AlpacaFarm-style cross-annotations (multiple independent raters per pair) and analogous multi-rater preference sets. Each datapoint i consists of a prompt x_i and two candidate completions; we set (y_i^w, y_i^ℓ) according to the majority label in the full pool of annotations for that pair, reserving the remaining annotations as an evaluation reservoir. The key requirement is that for each i we have enough independent rater votes to estimate a stable "ground-truth" preference rate

$$\alpha_i^\star \;\approx\; \frac{\#\text{wins for } y_i^w}{\#\text{total annotations for pair } i},$$

with uncertainty bands (e.g. Wilson intervals) reported to flag intrinsically ambiguous items. This produces a realistic mix of near-consensus items ($\alpha_i^\star$ near 0 or 1) and disagreement-heavy items ($\alpha_i^\star$ near 1/2).

Estimating c_i from a chosen reference policy. Given a fixed reference model π_{ref}, we compute

$$c_i \;=\; \log \frac{\pi_{\text{ref}}(y_i^\ell \mid x_i)}{\pi_{\text{ref}}(y_i^w \mid x_i)} \;=\; \log \pi_{\text{ref}}(y_i^\ell \mid x_i) - \log \pi_{\text{ref}}(y_i^w \mid x_i),$$

using standard token-level log-likelihood under teacher forcing. Because completion lengths vary, we will report both (i) raw log-likelihood ratios (as above) and (ii) length-normalized variants, e.g. dividing by total tokens, and check that qualitative conclusions are invariant. We then fix β (and vary it in sensitivity sweeps) to obtain thresholds $t_i = \sigma(\beta c_i)$. This step is operationally important: it turns "reference confidence" into a concrete notion of how much human preference probability is needed to overcome inertia.

Offline acquisition simulation: how we generate labels and update posteriors. To simulate the sequential purchase of labels, we treat the held-out annotations for each pair i as an empirical proxy for i.i.d. Bernoulli draws with success probability $\alpha_i^\star$. Concretely, when an acquisition policy requests a label for i, we sample (without replacement, when available) one rater vote from the held-out pool for that pair and record it as a Bernoulli outcome. Starting from a shared Beta prior $\alpha_i \sim \text{Beta}(a_0, b_0)$, we update

$$\alpha_i \mid \mathcal{D}_i \sim \text{Beta}(a_0 + w_i,\; b_0 + n_i - w_i), \qquad p_i(n_i) = \Pr(\alpha_i > t_i \mid \mathcal{D}_i),$$

where $p_i(n_i)$ is computed via the regularized incomplete beta function. We track both the posterior tail $p_i(n_i)$ and the posterior variance as functions of the acquired labels, since the index is fundamentally about buying information near the decision boundary.

Acquisition policies to compare (fixed K). We will compare a family of label allocation rules under the same total budget K:

1. *Uniform:* choose i uniformly at random.
2. *Uncertainty-only:* oversample pairs with posterior mean near 1/2, e.g. maximize $\mu_i(1-\mu_i)$ where $\mu_i = \mathbb{E}[\alpha_i \mid \mathcal{D}_i]$.
3. *High-c heuristic:* oversample large positive c_i (the "reference-confident against the winner" tail), ignoring posterior uncertainty.
4. *Index policy (ours):* sequentially choose

$$i_{k+1} \in \arg\max_i \Delta_i(n_i) \;=\; \arg\max_i \mathbb{E}\big[v_i\big(p_i(n_i+1) - p_i(n_i)\big)\big]\,,$$

   where the expectation is taken over the next Bernoulli label under the current posterior for α_i. In practice this expectation is a two-point mixture over win/loss outcomes, so it is cheap to compute.

For transparency, we will run the index both with uniform welfare weights ($v_i \equiv 1$) and with synthetic heavy-tailed weights (to emulate deployment frequency skew), because the welfare-weight channel is where governance and product priorities enter.

Offline metrics: what "better auditing" means in the simulation. We will report four classes of outcomes. First, the *posterior welfare objective* itself,

$$\widehat{W}_K \;=\; \sum_i v_i\, p_i(n_i),$$

which directly measures what the principal believes about coverage after spending K labels. Second, an *oracle welfare* computed using $\alpha_i^\star$ as ground truth,

$$W^\star \;=\; \sum_i v_i\, \mathbf{1}\{\alpha_i^\star > t_i\}, \qquad \text{and the audited estimate error } \big|\widehat{W}_K - W^\star\big|.$$

Third, *coverage and confidence* diagnostics: the mass of deployment weight for which the audit has become decisive,

$$\mathrm{ConfMass}_K(\epsilon) \;=\; \sum_i v_i\, \mathbf{1}\{p_i(n_i) \geq 1-\epsilon \text{ or } p_i(n_i) \leq \epsilon\},$$

which captures whether labels are resolving actionable questions rather than repeatedly sampling intrinsically ambiguous items. Fourth, *tail targeting* summaries that connect back to the mechanism: histograms of acquired labels over c_i, and the share of budget allocated to the slice $\{i : c_i \geq c_0,\ p_i(n_i) \in (\epsilon, 1-\epsilon)\}$, i.e. "high reference inertia and unresolved learnability."

Measuring "ranking flips" and the structure of overrides. To connect the audit to downstream behavior, we will compute a pairwise "override mass" that counts how often human preferences plausibly overturn the reference. The simplest diagnostic is the reference sign disagreement $\mathbf{1}\{c_i > 0\}$ (reference prefers y_i^ℓ) combined with learnability $\mathbf{1}\{\alpha_i^\star > t_i\}$ (humans are strong enough to override inertia). The weighted sum

$$\text{OverrideMass}^\star \;=\; \sum_i v_i \,\mathbf{1}\{c_i > 0\}\,\mathbf{1}\{\alpha_i^\star > t_i\}$$

identifies the welfare-relevant region where the pipeline must "flip" relative to π_{ref} to satisfy preferences. We then check whether acquisition policies preferentially resolve uncertainty on this region by reporting the analogous posterior quantity $\sum_i v_i \,\mathbf{1}\{c_i > 0\}\, p_i(n_i)$ as K grows.

Sensitivity sweeps over β and reference quality. We will run the full simulation across a grid of β values to test the central comparative-static claim: as β increases, value concentrates in large positive c_i slices because thresholds t_i rise, and the index should increasingly dominate uncertainty-only heuristics that ignore reference inertia. Separately, we will vary the reference itself to emulate shifts in reference quality and calibration: (i) swapping π_{ref} across model sizes/families; (ii) temperature scaling or logit smoothing before computing c_i (to damp overconfidence); and (iii) ensembling-based c_i (mean log-prob vs. log-mean-prob) to test whether the index gains are driven by calibration artifacts. The main object we expect to move is the *tail behavior* of the c_i distribution; the evaluation will explicitly report how the index performance correlates with statistics such as $\Pr(c_i > 0)$ and upper quantiles of c_i.

End-to-end check: does index-based acquisition improve actual preference optimization? Because our auditing model abstracts away many training dynamics, we will also run an end-to-end experiment on a smaller subset. For each policy and budget K, we build a training set by taking exactly the acquired labels (with multiplicity) and run a standard preference-optimization procedure (e.g. DPO) starting from a fixed initialization, holding all optimizer hyperparameters constant across policies. We then evaluate on a disjoint held-out set of preference comparisons with fresh annotations, reporting:

- *Win rate* of the trained policy against the reference and against the baseline policy on held-out comparisons.
- *Flip rate* on the specific subset with $c_i > 0$ (where the reference initially prefers y^ℓ), which is the regime where learning requires overcoming inertia.

- *Stability* metrics: variance across random seeds and across different rater-subset samplings, to ensure gains are not driven by lucky draws.

The point of this end-to-end check is not to perfectly validate the idealized threshold model, but to test whether the *ordering signal* provided by (c_i, posterior straddling) translates into measurable gains in realistic training.

Practicalities, failure modes, and what would falsify the story. We will pre-register two key falsifiable expectations. First, if the index is correct, then at moderate K it should achieve higher $\mathrm{ConfMass}_K(\epsilon)$ and higher estimated override mass than uniform sampling, while also reducing $|\widehat{W}_K - W^\star|$; failure here would indicate that the marginal-value computation $\Delta_i(n_i)$ is not aligned with information gain in real rater data. Second, index gains should *increase* with β and with heavier-tailed positive-c reference errors; if gains do not increase in these regimes, that would suggest either (i) c_i is too noisy a proxy for true reference inertia (e.g. due to length effects or scoring mismatch), or (ii) the pipeline is dominated by phenomena outside the model (e.g. generalization across datapoints, nonlocal parameter sharing, or systematic rater biases). In either case, the empirical results still provide a useful outcome: they tell us whether an auditable, reference-conditioned index is a meaningful organizing principle for label allocation, or whether we need richer primitives (e.g. clustered tasks, latent rater models, or training-aware value estimates) to make auditing predictive.

10. Discussion: auditability and governance in 2026. By 2026, preference-optimized models are increasingly embedded in products whose failure modes are not well summarized by average win rates. What matters operationally is whether the training pipeline can *reliably override* the reference policy on the specific slices of behavior where (i) humans strongly disagree with the reference, and (ii) that slice carries meaningful deployment weight. Our formalism is intentionally narrow—it compresses the downstream optimization into a threshold test $\alpha_i > t_i = \sigma(\beta c_i)$—but this compression has a governance upside: it yields quantities that an assurance team can compute, monitor, and report without re-running training for every audit question. The core claim is that $(c_i, \mathcal{D}_i, v_i)$ induces an *auditable interface* between model-centric artifacts (log-probabilities under π_{ref}) and human-centric uncertainty (posterior mass over α_i), and that this interface can be made legible to external stakeholders.

What a regulator (or internal assurance team) can compute. In settings where teams can log reference likelihoods and store preference-collection traces, three families of statistics become straightforward to compute and hard to fake post hoc.

(i) The distribution of reference inertia, via the c-profile. The object $c_i = \log \frac{\pi_{\text{ref}}(y_i^\ell|x_i)}{\pi_{\text{ref}}(y_i^w|x_i)}$ is not merely a training artifact; it measures how much probability mass the reference puts on the *dispreferred* completion relative to the preferred one, i.e. how far the pipeline must push against the reference to implement the preference. For governance, the important quantity is not a single mean but the *tail behavior* of c_i, because high positive c_i items are precisely where inertia makes learning brittle. An assurance report can therefore publish (possibly weight-adjusted) summaries such as

$$Q_{0.9}(c),\ Q_{0.99}(c),\ \Pr(c_i > 0), \quad \text{and} \quad \Pr(c_i > c_0 \text{ and } v_i \text{ large}),$$

as well as a welfare-weighted CDF

$$F_c(u) \ = \ \frac{1}{\sum_j v_j} \sum_i v_i \, \mathbf{1}\{c_i \le u\}.$$

These are not sufficient for safety, but they are a minimal transparency layer: they reveal whether the system relies on preference training to correct a small number of easy overrides or a long tail of high-inertia corrections.

(ii) Audited coverage, as decisive posterior mass rather than label count. Raw labeling spend is a poor proxy for assurance because labels can be wasted on intrinsically ambiguous comparisons. What can be reported instead is how much *deployment weight* has become audit-decided. A simple template is a weight-adjusted decisive-mass curve

$$\text{DecisiveMass}_K(\epsilon) \ = \ \sum_i v_i \, \mathbf{1}\{p_i(n_i) \ge 1 - \epsilon \text{ or } p_i(n_i) \le \epsilon\},$$

where $p_i(n_i) = \Pr(\alpha_i > t_i \mid \mathcal{D}_i)$. This statistic answers a governance-relevant question: "After spending K comparisons, for what fraction of what we care about do we have high confidence that the pipeline *can* (or *cannot*) overcome inertia?" Because ϵ is explicit, stakeholders can demand confidence levels appropriate to the domain (e.g. stricter in medicine than in entertainment).

(iii) Expected correction mass: how much welfare-relevant behavior requires overriding the reference. A central safety concern is whether preference optimization is mostly polishing already-good reference behavior or actually correcting systematic reference mistakes. In our abstraction, the correction-relevant region is $c_i > 0$ (reference favors y^ℓ) combined with learnability $\alpha_i > t_i$. Since α_i is unobserved, an auditable surrogate is the posterior expectation

$$\text{ECM}_K \ = \ \sum_i v_i \, \mathbf{1}\{c_i > 0\} \, p_i(n_i),$$

which we can interpret as *expected correction mass* under the audit posterior. Reporting ECM_K alongside $\sum_i v_i p_i(n_i)$ separates "coverage of learnable preferences" from "coverage of overrides." This distinction matters because a system can have high apparent preference alignment on easy items while still failing exactly where the reference is confidently wrong.

What should be reported, and how it can be gamed. A plausible 2026 assurance norm is a "reference-conditioned audit card" that reports the three objects above across a grid of β values (or, equivalently, across thresholds t_i). The reason is straightforward: β functions like an inertia knob, and many organizations will tune it for product stability, latency, or brand risk. A governance regime that only evaluates one β risks Goodharting: teams can pick β to look good on an audit while leaving high-inertia corrections unaddressed. A robust report would therefore include sensitivity curves

$$\beta \mapsto \mathrm{DecisiveMass}_K(\epsilon; \beta), \qquad \beta \mapsto \mathrm{ECM}_K(\beta),$$

and would flag regimes where small changes in β cause large drops in expected correction mass (an indicator of brittle reliance on borderline overrides).

At the same time, these metrics can be gamed if the organization can arbitrarily choose what pairs (y_i^w, y_i^ℓ) are surfaced to auditors. The remedy is procedural: auditors must sample x_i from *deployment-weighted* logs (to anchor v_i), and the organization must provide the *paired* candidates that the pipeline actually encounters (to anchor c_i). In other words, the governance surface is not just a metric suite; it is a data provenance constraint.

Safety implications and failure modes. The main safety-relevant failure mode in this framework is *illusory coverage*: high decisive mass on low-c regions combined with persistent uncertainty on high-c regions. This can happen even when overall win rate is high, because low-c items are cheap to resolve. A second failure mode is *calibration collapse* in c_i: if reference likelihood ratios are distorted by length effects, prompt formatting shifts, or log-probability miscalibration, then the threshold $t_i = \sigma(\beta c_i)$ ceases to track true inertia. Governance reports should therefore include basic calibration diagnostics, e.g. length-normalized c_i variants and stability of the c-profile under temperature scaling of π_{ref}. A third failure mode is *non-stationarity*: as the deployed system shifts the distribution of encountered prompts, both v_i and the effective α_i change, so historical audit coverage can become stale. This suggests an operational requirement: the audit index should be recomputed on a rolling basis, and "coverage over time" should be treated as a monitored quantity rather than a one-time certification.

Limitations of the single-threshold idealization. Our threshold model is a deliberate simplification of preference optimization. In reality, parameter sharing couples datapoints; learning on one slice may generalize (or anti-generalize) to another; and optimization dynamics can fail even when $\alpha_i > t_i$. From a governance perspective, this means the reported quantities should be interpreted as *pipeline capability under an idealized mechanism*, not as guarantees of realized behavior. The right use is comparative and

diagnostic: identifying where the pipeline is *least likely* to correct the reference absent additional evidence, and prioritizing those regions for more labels, better rater protocols, or targeted evaluations.

Extensions: multi-output ranking and richer training signals. Many production systems do not train on isolated pairs but on lists of candidates, tool-augmented trajectories, or multi-turn dialogues. The natural extension is to replace α_i with a structured preference object (e.g. a Plackett–Luce model over multiple outputs, or a latent utility function), and to replace c_i with a vector of reference log-odds that captures *which* alternative the reference prefers and by how much. The governance analog would report not just a scalar c-tail but a *confusion geometry*: where the reference concentrates probability among undesirable modes. Technically, this pushes the posterior tail probability $p_i(n_i)$ into higher-dimensional integrals, but the qualitative audit logic remains: label value concentrates near the reference-conditioned decision boundary, and welfare weights determine which boundaries matter.

Extensions: heterogeneous raters and institutional disagreement. In many domains, α_i is not a property of the world but of a rater population. If raters are heterogeneous, then a single Bernoulli parameter conflates moral uncertainty, expertise gaps, and demographic disagreement. For governance, this is not a nuisance; it is often the point. A more faithful model treats labels as drawn from a mixture (e.g. $\alpha_i^{(g)}$ per group g) or from a hierarchical rater model with rater-specific bias and noise. The audit surface then expands: instead of reporting $\Pr(\alpha_i > t_i \mid \mathcal{D}_i)$, one reports group-conditional learnability probabilities and a decomposition of decisive mass by population. This makes value conflicts explicit and enables a regulator to ask the correct question: "Which populations does the system have evidence of satisfying in the high-inertia override regime?" The downside is political as well as statistical: organizations must commit to a rater sampling frame, and audits become contested when stakeholder groups disagree.

Extensions: strategic judges and adversarial feedback. Finally, the assumption that labels are i.i.d. Bernoulli draws is fragile when raters have incentives (financial, ideological, or adversarial) to misreport, or when "raters" are themselves models. In 2026 this is not hypothetical: red-teaming markets, sybil attacks on feedback channels, and automated judging are all live. The natural extension is to model raters as strategic agents and treat α_i as endogenous to the mechanism (payments, auditing of raters, reputation systems). In that setting, the principal's control problem becomes two-layered: allocate budget across datapoints *and* invest in label integrity. The governance translation is that assurance reports should include not only coverage metrics but also *label provenance metrics*: rater diversity, fraud detection

rates, and robustness checks (e.g. whether ECM_K materially changes after removing a suspected rater cluster). The open problem is to integrate such integrity constraints into the same index framework without losing tractability.

Takeaway for governance. The central contribution of this line of work is not a new training objective but an accountability primitive: a way to say, in deployment-weighted terms, *where* preference optimization is expected to override the reference, *how confident* we are about those overrides, and *how that confidence depends* on inertia β and on the reference's own likelihood geometry. Even if the threshold abstraction is imperfect, it forces a discipline that aligns with governance needs: treating label budgets as scarce, treating "alignment" as coverage over high-stakes regions rather than average performance, and exposing the tails—the exact places where both safety and regulatory scrutiny concentrate.