SCREEN BEFORE YOU ADD: REDUCING CANDIDATE INFERENCE IN MODEL ENSEMBLES

Anonymous authors
Paper under double-blind review

ABSTRACT

As new models become available, existing ensembles must be updated to be competitive, making them intrinsically dynamic. Deciding whether to add a candidate model can require inference on an entire annotated dataset. Such a screening becomes prohibitive for models with costly inference especially when they ultimately add no value. Even under simple convex aggregation, relying on benchmark scores can be misleading as strong standalone performance does not guarantee ensemble improvement, and fitted ensemble weights can hurt test performance. In this work, we study candidate screening and show how early rejection can be achieved safely, exploiting residual geometry for stratification and rejecting the candidate when the best possible remaining predictions cannot achieve the necessary improvement. The key result that justifies this approach is that, under convex aggregation and squared or Brier loss, the population optimum improves if and only if a candidate induces a prediction change positively aligned with the residuals of the population-optimal existing ensemble. An optimistic quadratic program (QP) provides certified early rejection, while posterior-scenario sampling offers an alternative heuristic. We evaluate the approach by showing the trade-off between saved calls and screening accuracy, which refers to recovering the same accept/reject result as for the entire annotated dataset. We assess empirically whether full-dataset decisions predict candidate usefulness on a held-out test set. On two LLM panels, stratified screening with certified QP rejection saves 21–23% of candidate calls while keeping 100% screening accuracy. In our experiments, sampling posterior scenarios further increases call savings to approximately 37%, with a loss of less than one percentage point in screening accuracy.

1 INTRODUCTION

Ensemble methods combine predictions from multiple models to improve accuracy by exploiting differences in their errors (Breiman, 1996; Wood et al., 2023). Combining capabilities from different models is also an important element of modern LLM systems (Chen et al., 2025), this is achieved via direct aggregation, as in EVA and PackLLM (Xu et al., 2024; Mavromatis et al., 2024), or model coordination, as in Fugu (Sakana AI, 2026). Routing methods such as RouteLLM instead select a model for each input to reduce inference costs while maintaining response quality (Ong et al., 2025).

Maintaining such model pools raises the recurring **screening** question of whether a newly available model is worth adding. The rapid growth in foundation-model releases makes this question increasingly relevant (Maslej et al., 2024). We call the ensemble formed from the existing models the *incumbent ensemble*, borrowing terminology from AutoML, where "incumbent" denotes the best solution found so far (Hutter et al., 2011). We use a *screening set* of queries, annotations, and cached incumbent predictions to fit aggregation weights and compare losses with and without a candidate. Only candidate predictions require new inference, but full evaluation requires querying the entire set. For expensive models, this can make the screening decision costly even when the candidate ultimately adds no value.

Standalone performance does not settle this question. A candidate's value depends on the errors left by the existing ensemble and on how the models work together. A weaker model can correct some of those errors, while a stronger model can be redundant. Under convex aggregation, the refitted ensemble can assign the candidate zero weight, so its optimized loss on the screening set cannot

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

Table 1: **Effects of adding each model on 15 tasks.** Held-out top-1 accuracy; weights minimize screening-set Brier loss. Helped/hurt/unchanged give task counts; means give within-group ensemble accuracy changes (pp). Standalone accuracy and calls saved are in %. A perfect oracle skips harmful and unchanged tasks at zero decision cost. Details and size conventions: Appendix D.2.

	Original models (Each added to the other 7)								New models (Added to the original 8)			
	Qwen3	Llama 3.2	Gemma 4	Ministral 3	Phi-4 Mini	Granite 4.1	OLMo 3	Falcon-H1	SmolLM3	MiniCPM 4.1	GLM-4	Apertus
Size	4B	3B	E2B	3B	3.8B	3B	7B	7B	3B	8B	9B	8B
Standalone	78.14	65.60	68.35	69.73	71.21	73.62	70.10	79.98	67.34	74.16	79.91	59.17
Helped	7	2	4	4	2	5	5	9	4	5	8	1
Mean gain	+1.47	+1.41	+0.94	+1.41	+0.47	+0.79	+0.75	+3.38	+0.94	+1.46	+2.45	+0.31
Hurt	4	5	1	5	2	2	2	3	2	3	2	2
Mean loss	−1.02	−0.46	−1.56	−0.50	−0.42	−0.47	−0.94	−0.52	−0.47	−0.42	−0.47	−0.78
Unchanged	4	8	10	6	11	8	8	3	9	7	5	12
Calls saved	53.3	86.7	73.3	73.3	86.7	66.7	66.7	40.0	73.3	66.7	46.7	93.3

increase. Nevertheless, it may result in no gain and weights fitted on the screening set can perform worse on held-out examples. Table 1 illustrates this across twelve LLMs and 15 task families. Qwen3 improves held-out accuracy by a mean of 1.47 percentage points on seven tasks, but hurts four and leaves four unchanged. Every tested model hurts at least one task, including the largest and best standalone performer. An oracle that already knows which tasks benefit in held-out accuracy can skip all candidate calls on the others, avoiding 40.0–93.3% of screening-set calls across models.

We study screening under convex aggregation with squared or multiclass Brier loss (Brier, 1950). Adding a candidate allows all ensemble weights to change. This setting enables an exact residual-alignment characterization of population improvement and a convex QP for safe early rejection on a finite screening set. Convex aggregation under squared loss is an established setting in the theoretical literature (Lecué, 2013). Practical precedents for this setup include MSE-optimized LLM ensembles for well-being score prediction (Ravenda et al., 2025) and Brier-optimized combinations of LLM forecasts with prediction-market probabilities (Alur et al., 2025).

Our goal is to recover the full-screening-set keep/reject decision at a 1% relative-loss improvement threshold using fewer candidate calls. Cached residuals and prediction bounds guide stratification, grouping queries by uncertainty in their alignment contributions (Figure 1). An optimistic quadratic program certifies rejection when even the most favorable unobserved predictions cannot achieve the required gain. A posterior-scenario extension models these predictions to trade some screening accuracy for additional call savings. We measure calls saved and *screening accuracy*: the percentage of keep/reject decisions matching evaluation on the full screening set.

Experiments cover 35 regression tasks and two LLM panels. On the LLM panels, certified screening saves 21–23% of candidate calls relative to full evaluation while maintaining 100% screening accuracy. Posterior scenarios raise savings to about 37% while retaining over 99% screening accuracy. We separately check whether full-set decisions predict improvement on held-out examples.

Our contributions are: **(i)** an exact characterization of candidate improvement under population-optimal convex aggregation with squared or Brier loss; **(ii)** a screening method combining residual-based stratification with certified early rejection to reduce candidate calls; and **(iii)** a posterior-scenario extension, evaluated alongside certified screening on regression and LLM tasks to measure the tradeoff between screening accuracy and call savings.

2 Related work

Aggregation. Stacking learns how to combine predictors (Wolpert, 1992; Breiman, 1996), and statistical aggregation studies excess risk for model selection, linear combinations, and convex combinations (Tsybakov, 2003). These lines of research set the foundations for modern aggregation techniques answering **how** to aggregate models. In our work, we ask how sample-efficiently we can answer **when** one supplied candidate can improve an existing convex ensemble.

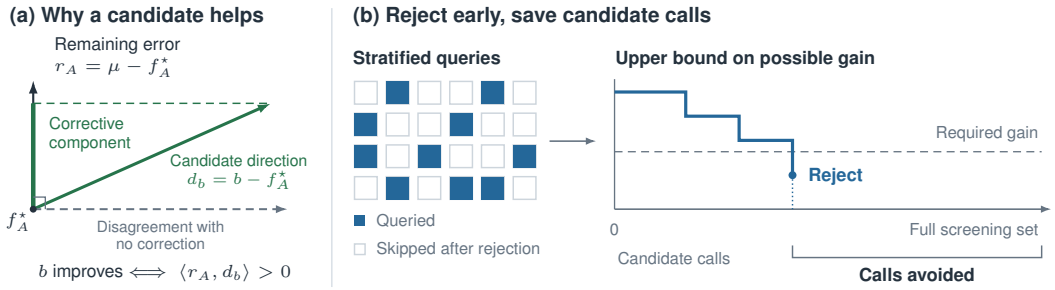

Figure 1: **Candidate screening from cached predictions.** (a) Under squared loss, positive residual alignment characterizes improvement of the population-optimal convex ensemble. (b) Rows represent strata; filled cells are queried and empty cells are skipped. Rejection is certified when the optimistic gain bound falls below the required improvement.

Residual correction. Our screening approach is based on residual geometry. Residuals are at the core of methods like gradient boosting, where they guide how to construct new components (Friedman, 2001). In our setting, the candidate already exists and evaluating its predictions is costly. Prior work on ensemble diversity relates disagreement to individual errors and the aggregation rule (Brown et al., 2005; Wood et al., 2023). Our projection criterion states exactly when this combination of quality and disagreement permits an improvement.

Sample-efficient evaluation. Active testing selects examples to estimate risk with fewer labels (Kossen et al., 2021). Compact LLM benchmarks use selected subsets and psychometric structure to approximate full-benchmark performance (Maia Polo et al., 2024), while surrogate uncertainty can guide approximate Neyman allocation (Liu et al., 2026b). The main distinction with our work is the evaluation target and the available prior information. In our setting, labels and incumbent predictions are available, while the missing information is the candidate's output. This is motivated by the practical scenario where a practitioner has already trained an aggregation but new models are released. We use classical stratified estimation and allocation (Cochran, 1977) to estimate its residual alignment. The theory controls this estimate, and the experiments assess agreement with the ensemble decision from the full screening set.

Safe screening. Safe screening uses optimization bounds to discard options without changing the target solution (Ndiaye et al., 2015). Dantas & Gribonval (2019) make elimination safe under incomplete knowledge of the predictor vectors identifying inactive variables in ℓ_1-regularized regression. Hutter et al. (2009) leverage the logic of stopping when even the best remaining outcome cannot change the decision, so the shared principle with our approach is bounding what the remaining evaluation could achieve. In our setting, rejection must remain valid after full refitting of ensemble weights, so we bound both the effect of unobserved candidate predictions and the best improvement achievable by the aggregation weights. Statistical bounds for early termination provide an alternative to considering every admissible completion, while achieving looser statistical guarantees. Maron & Moore (1993) use confidence bounds to reject candidates before completing their evaluation, and confidence sequences provide bounds that remain valid across repeated checks as new data arrive (Howard et al., 2021). We take inspiration to use probabilistic evidence and design a posterior-scenario heuristic modeling unobserved candidates to trade off safety for earlier rejection.

3 Problem Setup and Theory

We first characterize when a candidate improves the best ensemble formed by the incumbent models. The result captures the candidate's value after population-level weight reoptimization.

Let $A = \{a_1, \ldots, a_m\}$ be fixed predictors, b be a candidate, and $(X, Y) \sim P$ and P_X denotes the marginal distribution of X. Assume squared loss, $\mathbb{E}[Y^2] < \infty$, $a_j, b \in L^2(P_X)$, and unregularized convex aggregation. Here, $L^2(P_X)$ is the space of functions f with finite expected squared magnitude, $\mathbb{E}[f(X)^2] < \infty$. For $\Delta_m = \{w \in \mathbb{R}^m : w \geq 0,\ \mathbf{1}^\top w = 1\}$, an aggregate is $f_{A,w} = \sum_j w_j a_j$.

Let $\mathcal{C}_A = \mathrm{conv}(A)$ and $\mathcal{C}_{A+b} = \mathrm{conv}(A \cup \{b\})$, where $\mathrm{conv}(A)$ is the set of all convex combinations of predictors in A. For any predictor $f \in L^2(P_X)$, define $R(f) = \mathbb{E}[(Y - f(X))^2]$ and the corresponding population minimizers by

$$f_A^\star = \arg\min_{f \in \mathcal{C}_A} R(f), \qquad f_{A+b}^\star = \arg\min_{f \in \mathcal{C}_{A+b}} R(f). \tag{1}$$

The candidate's population value is $\Delta_b^\star = R(f_A^\star) - R(f_{A+b}^\star) \geq 0$. Their risk difference measures the improvement available in the expanded convex hull. When weights are learned from finite data, held-out performance also depends on estimation error (Appendix A).

Under squared loss, this comparison has a geometric form. We write $\mu(x) = \mathbb{E}[Y \mid X = x]$ and use $\langle u, v \rangle = \mathbb{E}[u(X)v(X)]$ with its induced norm. Since $R(f) = \mathbb{E}[\mathrm{Var}(Y \mid X)] + \|\mu - f\|^2$, $f_A^\star$ is the $L^2(P_X)$ projection of μ onto $\mathcal{C}_A$, which is the convex combination of incumbent predictors closest to μ in expected squared distance, $\mathbb{E}[(\mu(X) - f(X))^2]$. We define the residual, candidate direction, and alignment by $r_A = \mu - f_A^\star$, $d_b = b - f_A^\star$, $c_b = \langle r_A, d_b \rangle$. The projection optimality gives $\langle r_A, a - f_A^\star \rangle \leq 0$ for every $a \in \mathcal{C}_A$. Reweighting the existing models cannot further reduce population risk. The next theorem characterizes when adding b can.

Theorem 1 (Exact residual-alignment criterion)**.** *Under population-optimal convex aggregation with squared loss,* $\Delta_b^\star > 0 \iff c_b = \langle r_A, b - f_A^\star \rangle > 0$.

Proof. If $c_b > 0$, the feasible segment $f_\lambda = f_A^\star + \lambda d_b$ has gain $R(f_A^\star) - R(f_\lambda) = 2\lambda c_b - \lambda^2 \|d_b\|^2 > 0$ for sufficiently small $\lambda > 0$. Conversely, if $c_b \leq 0$, every $g = (1-\lambda)a + \lambda b \in \mathcal{C}_{A+b}$, with $a \in \mathcal{C}_A$ and $\lambda \in [0, 1]$, satisfies $\langle r_A, g - f_A^\star \rangle = (1 - \lambda) \langle r_A, a - f_A^\star \rangle + \lambda c_b \leq 0$. Thus $f_A^\star$ satisfies the projection condition on the expanded hull and remains optimal. □

The theorem identifies which part of a candidate's disagreement can improve the ensemble: its component along the residual of the population-optimal incumbent. Disagreement orthogonal to the incumbent residual cannot improve the population-optimal ensemble. When alignment is positive, reoptimizing all weights can improve on the direct move toward the candidate. When alignment is nonpositive, reoptimization cannot produce a gain.

Labels, standalone quality, and disagreement. The criterion involves the unknown conditional mean μ. To connect it to observable data, we express the same alignment using labels. Conditional expectation gives $c_b = \mathbb{E}[(Y - f_A^\star(X))(b(X) - f_A^\star(X))]$. The noise term has zero expectation conditional on X, so the score does not require the unknown regression function explicitly. The population-optimal incumbent predictor $f_A^\star$ and the expectation still have to be estimated in practice. The same score also connects standalone quality with disagreement. To highlight this we rewrite c_b using $q_b = \|b - f_A^\star\|^2$ and expanding $R(b)$ which gives $c_b = 1/2\ (R(f_A^\star) - R(b) + q_b)$. This identity explains why standalone quality by itself cannot determine a candidate's value. In particular, $0 < c_b < q_b/2$ describes a candidate that is worse alone than the incumbent aggregate but improves it at a smaller convex weight. Appendix A derives the optimal gain on the direct segment.

Multiclass Brier loss. The argument relies on squared-error geometry, which is also preserved by multiclass Brier loss for probability predictions. Let $Y \in 1, \ldots, K$ and let $p : \mathcal{X} \to \Delta_K$ denote a probability predictor, where Δ_K is the K-class probability simplex. Define $R(p) = \mathbb{E}[\|e_Y - p(X)\|_2^2]$, where $e_Y \in \mathbb{R}^K$ is the one-hot vector corresponding to class Y. For square-integrable vector-valued functions, use the inner product $\langle u, v \rangle = \mathbb{E}[u(X)^\top v(X)]$. The same projection argument then gives $R(p_A^\star) > R(p_{A+b}^\star) \iff \mathbb{E}[(e_Y - p_A^\star(X))^\top (p_b(X) - p_A^\star(X))] > 0$, where p_b is the candidate probability predictor and $p_S^\star$ denotes the population-optimal convex aggregate of the probability predictors in S. This is the loss and alignment used in the LLM experiments.

4 METHOD

The previous section characterized candidate improvement at the population optimum. We now turn to finite data, aiming to recover the full-screening-set decision with fewer candidate calls. Using labels and incumbent predictions, we construct a residual-informed stratified query path before observing candidate outputs. Along this path, an optimistic QP certifies rejection when no completion of the unseen predictions consistent with the known bounds can achieve the required improvement.

4.1 SCREENING TARGET AND QUERY MODEL

The screening target is the decision obtained by fitting both ensembles on the complete screening set. For regression, let $y = (y_1, \ldots, y_n)^\top \in \mathbb{R}^n$, while for Brier loss, we instead use the concatenated one-hot target vector. For a screening set of n rows (x_i, y_i), we identify each predictor by its vector with elements $f_i = f(x_i)$ and $b_i = b(x_i)$. We also use the empirical inner product $\langle u, v \rangle_n = n^{-1} \sum_i u_i^\top v_i$, and its induced norm. For regression and multiclass Brier loss, respectively, we use $R_n(f) = \frac{1}{n} \sum_i (y_i - f_i)^2$, $R_n(p) = \frac{1}{n} \sum_i \|e_{y_i} - p_i\|_2^2$, where $p_i = p(x_i)$. Let $f^\star_{A,n}$ and $f^\star_{A+b,n}$ minimize empirical risk over the incumbent and augmented convex hulls. Define $r_{A,n} = y - f^\star_{A,n}$, $c_{b,n} = \langle r_{A,n}, b - f^\star_{A,n} \rangle_n$, $\Delta^\star_{b,n} = R_n(f^\star_{A,n}) - R_n(f^\star_{A+b,n})$, where n indicates fitting on the complete screening set. For the relative-gain criterion we assume $R_n(f^\star_{A,n}) > 0$ and call a candidate material on the screening set when

$$\Delta^\star_{b,n} \geq \tau_n, \qquad \tau_n = \gamma R_n(f^\star_{A,n}), \qquad \gamma = 0.01. \tag{2}$$

This finite-set target is distinct from population improvement. Whether adding the candidate also achieves the required loss reduction on held-out data is tested separately, not assumed. Held-out results never guide querying or stopping. Only the candidate values b_i are initially missing. For scalar regression, querying row i reveals its prediction and the alignment contribution $z_i = r_{A,n,i}(b_i - f^\star_{A,n,i})$. Here, $r_{A,n,i}$ and $f^\star_{A,n,i}$ denote the residual and fitted incumbent ensemble prediction on screening example i.

4.2 BUILDING A CANDIDATE-INDEPENDENT QUERY PATH

Before querying the candidate, we can bound how much each row may contribute to its residual alignment. Suppose each unseen prediction lies in a known interval $[L_i, U_i]$, its contribution then lies in the envelope $[\ell_i, u_i]$, where $\ell_i = \min\{r_{A,n,i}L_i, r_{A,n,i}U_i\} - r_{A,n,i}f^\star_{A,n,i}$, and $u_i = \ell_i + |r_{A,n,i}|(U_i - L_i)$. For multiclass Brier loss, the probability simplex supplies the prediction bounds. Set $r_i = e_{y_i} - p^\star_{A,i}$, where $p^\star_{A,i}$ is the ith-row of the incumbent probability aggregate fitted on the screening set. The unknown row contribution is $z_i = r_i^\top (p_{b,i} - p^\star_{A,i})$, with $p_{b,i} = p_b(x_i)$, whose exact extrema over that simplex are $\ell_i = \min_k r_{ik} - r_i^\top p^\star_{A,i}$ and $u_i = \max_k r_{ik} - r_i^\top p^\star_{A,i}$, where K_i is the number of valid classes for row i, then we have $k = 1, \ldots, K_i$ and r_{ik} is coordinate k of r_i. One query reveals the complete candidate probability vector on that row.

Allocation and estimation. We use these known ranges to allocate queries before seeing any candidate outputs. We sort rows by envelope width and form four strata as equal in size as possible. For stratum h, let N_h be its size, $W_h = N_h/n$, and $R_h = \max_{i \in h} u_i - \min_{i \in h} \ell_i$ its outer-envelope range. The target $\pi^\star_h$ assigns more queries to strata with wider ranges. We mix it equally with proportional allocation to obtain $\widetilde{\pi}_h$, retaining a baseline share of queries for every stratum

$$\pi^\star_h = \frac{W_h R_h}{\sum_g W_g R_g}, \qquad \widetilde{\pi}_h = \tfrac{1}{2} W_h + \tfrac{1}{2} \pi^\star_h. \tag{3}$$

Thus, half of the allocation follows stratum size, while the other half favors strata with greater possible alignment variation. If all ranges vanish, set $\pi^\star_h = W_h$ for every stratum h. To approximately follow the target fractions, we randomly permute rows within each stratum and query one from each initially. We then repeatedly select the nonexhausted stratum with largest $\widetilde{\pi}_h^2/[n_h(n_h+1)]$ and query its next permuted row, where n_h is its current query count. This produces a single query ordering, independent of candidate outputs, whose first B rows are used at budget B. At budget $B = \sum_h n_h$, the alignment estimate is $\widehat{c}_B = \sum_h W_h \overline{z}_h$, where $\overline{z}_h$ is the mean of the queried alignment contributions in stratum h.

Informal Proposition (Allocation guarantee, see Proposition 3)**.** *For equal-sized strata, $\pi^\star$ minimizes the worst-case variance of the alignment estimate under the specified stratum bounds, allowing fractional query counts and assuming no stratum is fully queried. The mixture $\widetilde{\pi}$ keeps the alignment-error bound no larger than that of proportional allocation and at most $\sqrt{2}$ times that of $\pi^\star$, provided the target fractions are realized exactly.*

Each sampled row receives weight N_h/n_h in the refit. These weights correct for unequal sampling across strata. For any fixed predictor f, $\widehat{R}_B(f)$ is an unbiased estimator of $R_n(f)$. Appendix B

states the finite-population, concentration, and weight bounds; Appendix B.3 gives their proofs and the integer scheduler argument.

No fixed nonproportional allocation uniformly outperforms uniform sampling when alignment contributions are unrestricted: for any such allocation, there exist contribution patterns for which uniform sampling has lower variance (Proposition 2). Our guarantees concern estimation of residual alignment under the specified stratum bounds. The keep/reject decision instead depends on the loss after refitting the ensemble weights, so a more accurate alignment estimate does not automatically imply a more accurate decision. We therefore test empirically whether the proposed allocation improves agreement with decisions from the full screening set.

Incumbent loss as an alternative score. Incumbent loss gives another way to prioritize rows using information already in the screening set. The second scheduler uses the known row loss, with $\eta_i = r_{A,n,i}^2$ for scalar squared loss and $\eta_i = \|e_{y_i} - p^\star_{A,i}\|_2^2$ for multiclass Brier loss. The following relation explains why this simpler score tracks the available alignment on each row.

Proposition 1 (Incumbent loss tracks alignment headroom)**.** *For scalar predictions in a common interval $[L, U]$, the width $w_i = u_i - \ell_i$ satisfies $w_i = (U - L)\sqrt{\eta_i}$. For the Brier simplex envelope, $u_i = \eta_i$, $\sqrt{\eta_i} \le w_i \le \sqrt{2\eta_i}$. For binary rows, $\eta_i = w_i^2/2$ exactly.*

Appendix B.5 gives the proof. Loss and width rank regression rows identically. For Brier loss, incumbent loss specifies the maximum positive alignment and bounds its width. The loss scheduler sorts by η_i and feeds pseudo-intervals $[0, \eta_i]$ into the same four-stratum allocator. These intervals are scheduling scores, not alignment bounds. They can change the allocation even when the rankings agree. The stopping rule always uses genuine prediction bounds.

4.3 Safe rejection along the query path

Safe rejection asks whether even the most favorable unseen predictions could make the candidate useful. After a queried prefix, we reject only if no compatible completion can attain τ_n. For scalar regression, let $d = n$, $y \in \mathbb{R}^d$, and $\mathbf{A} \in \mathbb{R}^{d \times m}$ be the matrix whose columns contain the incumbent predictions. Let $\mathcal{I}_q \subseteq \{1, \dots, n\}$ denote the indices of screening examples queried after q candidate calls. The completion box is

$$\mathcal{B}_q = \{v \in \mathbb{R}^d : v_i = b_i \ (i \in \mathcal{I}_q), \quad L_i \le v_i \le U_i \ (i \notin \mathcal{I}_q)\}. \tag{4}$$

Assume finite prediction bounds, so the box is compact and contains the true candidate. We obtain the most optimistic augmented risk by choosing both the completion and the ensemble weights:

$$\begin{aligned} \rho_q^{\mathrm{box}} = \min_{\alpha,\beta,t} \ & \frac{1}{d}\|y - \mathbf{A}\alpha - t\|_2^2 \\ \text{subject to } \alpha \ge 0, \quad & \beta \ge 0, \quad \mathbf{1}^\top \alpha + \beta = 1, \\ & t_i = \beta b_i && (i \in \mathcal{I}_q), \\ & \beta L_i \le t_i \le \beta U_i && (i \notin \mathcal{I}_q). \end{aligned} \tag{5}$$

$\alpha \in \mathbb{R}^m$ contains incumbent weights, β is the candidate weight, and $t = \beta v \in \mathbb{R}^d$ is the weighted candidate completion. The substitution $t = \beta v$ makes the constraints linear, giving a convex QP with feasible prediction set $\mathrm{conv}(\{a_1, \dots, a_m\} \cup \mathcal{B}_q)$. For a complete vector v, write its gain on the screening set as $\Delta_n(v) = R_n(f^\star_{A,n}) - \min_{g \in \mathrm{conv}(A \cup \{v\})} R_n(g)$, and define $U_q = \max_{v \in \mathcal{B}_q} \Delta_n(v)$.

Theorem 2 (Complete safe rejection for a fixed completion set)**.** *The exact optimistic QP satisfies $U_q = R_n(f^\star_{A,n}) - \rho_q^{\mathrm{box}}$. For $\tau_n > 0$, rejecting when $U_q < \tau_n$ is safe and complete relative to $\mathcal{B}_q$: otherwise some compatible completion has gain at least τ_n. Every zero-error deterministic rejection rule based only on the same queried values and completion set therefore has a rejection region contained in the exact QP's rejection region.*

Proof. The perspective formulation minimizes augmented risk jointly over ensemble weights and $v \in \mathcal{B}_q$. Subtracting this joint minimum from incumbent risk proves the identity. Compactness and continuity ensure the maximum compatible gain is attained. If $U_q < \tau_n$, no compatible candidate is material; otherwise an attaining material candidate agrees with every observation, so a zero-error rule cannot reject it. □

As we show next, the exact QP certificate rejects the earliest among all safe rules after the same queries and considers the same possibilities for the unobserved predictions.

Corollary 1 (Earliest safe rejection on a fixed path)**.** *Among rules inspecting the same nested path at the same checkpoints, the exact QP rejects at the first checkpoint where any zero-error rule using the declared completion set can reject.*

Each query shrinks $\mathcal{B}_q$, so the maximum compatible gain U_q can only decrease. Applying the theorem at each checkpoint gives the corollary. Its comparison holds for the same path and checkpoints; it does not establish that the chosen path is query-optimal. Early acceptance would require a guarantee for *every* completion and leads to a different max–min problem which we do not study here (Appendix C.1). Thus, we rely on the refit at the budget or census for positive acceptance decisions.

Conservative numerical evaluation & budget fallback. A dual lower bound on the optimistic QP loss gives an upper bound on the candidate's best achievable improvement. This allows certified rejection without solving the QP exactly, although rejection may occur later than with the exact solution. Appendix D.4 describes the numerical safety margin and implementation details. If no certificate rejects before budget B, we refit both ensembles on the queried prefix using post-stratification weights. Assuming positive fitted incumbent loss, we keep the candidate if the relative weighted loss reduction is at least 1%. This fallback recovers the reference decision when the full screening set is queried, but may disagree at smaller budgets; only early rejection is certified.

5 Experiments

We evaluate whether screening can reproduce the keep/reject decision from the full screening set while avoiding candidate calls. We first compare sampling rules at fixed budgets, then hold the query path fixed to measure the additional savings from early rejection. On the same three panels, we also test how posterior scenarios trade screening accuracy for further savings. Separately, we assess whether the full-screening-set decision predicts candidate usefulness on held-out data and quantify the computational overhead of the stopping procedures.

5.1 Experimental setup and evaluation

Datasets and models. We evaluate on 35 regression datasets from OpenML-CTR23 (Fischer et al., 2023) and 15 language understanding, reasoning, and classification tasks presented in multiple-choice form (see Appendix D.2 for dataset references) using the same 12 LLMs listed in Table 1.

For regression, the four candidate families are ridge regression, random forest, histogram gradient boosting, and multilayer perceptrons. For each dataset and ten training seeds, a candidate from one family is evaluated against an ensemble built from the other three, giving $35 \times 4 \times 10 = 1{,}400$ cases.

The LLM evaluation contains two separate tasks. In the *unseen-task* evaluation, each of the eight original models is added to the other seven on 15 task families held out from policy development, giving 120 cases. In the *unseen-model* evaluation, each of four new-family candidates is added to the fixed original eight on those same tasks, giving 60 cases. Regression screening sets contain 129-3,000 examples, while each LLM task has 192 screening examples. Both settings use separate held-out data. Appendix D lists the tasks, model configurations, and data splits.

Metrics. The reference decision evaluates the candidate on the complete screening set and keeps it if refitting the ensemble reduces MSE or Brier loss by at least 1 percent (Equation (2)). We measure two quantities: *screening accuracy*, which is the percentage of decisions matching this reference where each task has equal weight, and *calls saved*, which measures the fraction of total candidate calls avoided, rather than the mean percentage saved per task. We average repeated runs within cases and candidate/seed results within tasks. We report *relative overhead*: stopping computation time as a percentage of estimated inference time saved, using Qwen3 4B one-token timings (Appendix D.5).

Held-out evaluation. We separately assess whether the reference decision predicts usefulness on unseen examples. We freeze the incumbent and augmented weights fitted on the complete screening set and label a candidate useful if the frozen augmented ensemble reduces held-out loss by at least

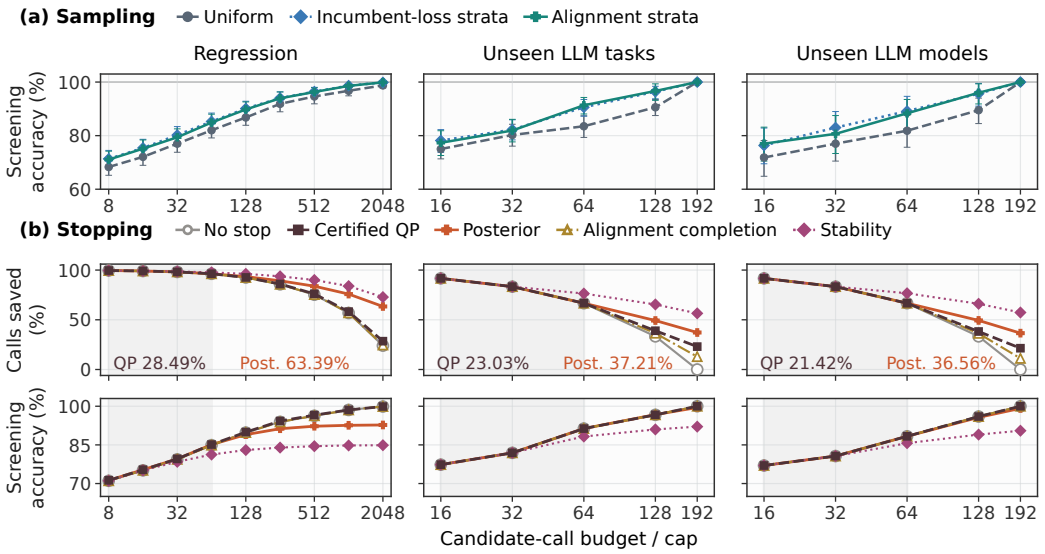

Figure 2: **(a) Screening accuracy at fixed query budgets.** Uniform sampling is compared with incumbent-loss and alignment stratification, with early stopping disabled. Bars show pointwise 95% hierarchical-bootstrap intervals. Regression budgets are capped at each screening set's size. **(b) Call savings and screening accuracy.** Top: calls saved relative to the complete screening set. Bottom: screening accuracy. The gray no-stop baseline shows savings from the budget cap alone. Posterior scenarios are uncertified and follow QP in the shaded region, through 64 calls. Curves show replay point estimates.

one percent relative to the frozen incumbent ensemble. The reference decisions agree with these held-out labels in 89.29% of regression cases, 85.00% of unseen-task cases, and 83.33% of unseen-model cases. This check evaluates the usefulness of the full-screening-set reference, as held-out predictions never enter querying or stopping. Appendix E.5 provides further diagnostics.

Comparison protocol. We replay cached predictions over ten runs per case, revealing only requested examples. Budgets reach 2,048 regression or 192 LLM calls, capped by screening-set size. Sampling comparisons disable stopping; stopping comparisons share the query order, rejection checks, and fallback. A no-stop baseline isolates savings from early rejection. Appendix D.3 details the paired bootstrap intervals.

5.2 Fixed-budget sampling

We first ask whether using residual information to choose queries improves screening accuracy at a fixed inference budget. Early stopping is disabled to isolate the effect of sampling.

Figure 2 (a) shows that both proposed sampling strategies consistently improve mean screening accuracy over uniform sampling across all three panels and evaluated query budgets, with all methods agreeing when the complete screening set is queried. Neither variant consistently outperforms the other. At 64 candidate calls, the two variants improve screening accuracy over uniform sampling by 2.9–3.3 percentage points in regression and 6.5–7.8 points across the two LLM panels. Residual information therefore helps identify informative queries: with the same number of candidate calls, screening more reliably recovers the decision from the complete screening set.

These findings complement the theoretical analysis. Proposition 2 shows that no fixed nonproportional allocation improves alignment-estimation variance over uniform sampling for every possible contribution population. Moreover, estimation guarantees alone do not ensure better decisions after refitting ensemble weights. Figure 2 (a) provides empirical evidence that both sampling variants help with this decision on the evaluated tasks. The variance analysis in Appendix B.4 further shows that most of the reduction in alignment-estimation variance comes from allocating queries unequally across strata, beyond the benefit of forming strata alone.

5.3 CERTIFIED REJECTION

We next ask how many additional calls early rejection can save along the same alignment-stratified query order. Alongside QP, we compare alignment completion, a simpler certified baseline that bounds each unobserved alignment contribution by its maximum. It turns this optimistic alignment into an upper bound on the gain after ensemble refitting, whereas QP optimizes jointly over the missing predictions and ensemble weights. We also compare two-negative stability, a heuristic baseline that rejects after two successive checks indicate insufficient improvement and is motivated by early-stopping criteria in NN training and active learning (Prechelt, 1998; Altschuler & Bloodgood, 2019).

Figure 2(b) shows that QP saves more calls than alignment completion at the same screening accuracy. On large budgets, QP saves 21–23% of candidate calls on the two LLM panels, compared with 11–13% for alignment completion, while both maintain 100% screening accuracy. These budgets cover the complete LLM screening sets, so all savings come from early rejection. The comparison shows that accounting for ensemble refitting in the optimistic bound enables earlier safe rejection.

In regression, QP saves 28.5% of calls with 99.89% screening accuracy, adding 4.5 percentage points of savings beyond the 24.0% provided by the budget cap alone. QP makes no incorrect early rejections on these runs; the remaining regression disagreements arise from decisions made when the budget is exhausted. Two-negative stability saves more calls, but incorrectly rejects candidates that complete evaluation would keep, reducing screening accuracy. Certified QP has a relative overhead of only 0.34%.

5.4 POSTERIOR RELAXATION

Certified QP rejection must consider even highly favorable remaining predictions, regardless of how plausible they are given the outputs observed. The posterior extension uses the candidate predictions already queried, together with the known incumbent predictions and labels, to model the remaining outputs. We sample complete prediction scenarios, keeping observed outputs fixed, and refit the ensemble weights in each scenario. Rejection occurs when an upper confidence bound on the modelled probability of achieving the required improvement falls below δ. This permits earlier rejection at the risk that the actual remaining predictions are more favorable than the model suggests.

At $\delta = 0.05$ and the largest budgets, Figure 2 (b) shows call savings of approximately 37% on both LLM panels, with screening accuracy falling by less than one percentage point relative to certified QP. In regression, savings increase to 63.4%, while screening accuracy falls from 99.89% to 92.73%, showing a larger accuracy cost. The confidence bound accounts for simulation error under the predictive model; it does not guarantee the rate of incorrect rejections in actual screening. To avoid basing posterior rejection on too few observed candidate outputs, we enable it only after 64 candidate calls; earlier rejection still uses the certified QP. Appendix F gives the predictive models, the scenario construction, and the sensitivity to the rejection threshold. Posterior scenarios increase relative overhead to approximately 3.7%.

6 CONCLUSIONS

Our approach reduces the inference needed to decide whether a costly model, such as an LLM, should join an ensemble. The theory and experiments show how information about the existing ensemble can guide this evaluation. Both sampling variants improve mean screening accuracy at partial budgets, while the optimistic QP certifies safe early rejection. Together, they reduce evaluation cost by choosing informative queries and stopping when the candidate cannot achieve the required gain. Full-screening-set decisions also agree substantially with candidate usefulness on held-out data, supporting them as a practical screening target while keeping generalization a separate question.

The early rejection can be conservative because it must consider every admissible set of unobserved predictions, including highly favorable ones. The posterior extension uses observed outputs to model plausible remaining predictions, empirically achieving greater savings at the cost of some incorrect rejections. Future work should establish probabilistic guarantees for these errors under explicit predictive-model assumptions and account for repeated checks. This would let practitioners balance inference savings against an acceptable probability of rejecting a useful candidate.

7 AI DISCLOSURE

In this work we used generative AI tools to help in: **(i)** checking and assisting algebraic proof steps after the theorem statements and proof strategies had been developed by the authors, **(ii)** implementation of the code for numerical evaluation, **(iii)** realizing figures, **(iv)** edit the text to improve readability, **(v)** support in defining relevant literature.

We have **not** used generative AI tools for: **(i)** help develop theoretical models or conceptual frameworks, **(ii)** provide critical ingredients for proving mathematical claims, **(iii)** propose or refine hypotheses, **(iv)** interpret results, **(v)** design or provide feedback on research methodology or experiments.

We have reviewed all AI-assisted work: **(i)** the validity of theoretical proofs was validated by 2 authors, **(ii)** the LLM-generated code was verified by 1 author, **(iii)** the correctness of generated figures was validated, **(iv)** the text was reviewed by all authors, **(v)** we verified the papers found via LLM-supported literature review. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Michael Altschuler and Michael Bloodgood. Stopping active learning based on predicted change of F measure for text classification. In *2019 IEEE 13th International Conference on Semantic Computing (ICSC)*, pp. 47–54. IEEE, 2019. doi: 10.1109/ICOSC.2019.8665646. URL https://arxiv.org/abs/1901.09118.

R. Alur, B. C. Stadie, D. Kang, R. Chen, M. McManus, M. Rickert, T. Lee, M. Federici, R. Zhu, D. Fogerty, H. Williamson, N. Lozinski, A. Linsky, and J. S. Sekhon. AIA forecaster: Technical report. *arXiv preprint arXiv:2511.07678*, 2025. URL https://arxiv.org/abs/2511.07678.

Elie Bakouch, Loubna Ben Allal, Anton Lozhkov, Nouamane Tazi, Lewis Tunstall, Carlos Miguel Patiño, Edward Beeching, Aymeric Roucher, Aksel Joonas Reedi, Quentin Gallouédec, Kashif Rasul, Nathan Habib, Clémentine Fourrier, Hynek Kydlicek, Guilherme Penedo, Hugo Larcher, Mathieu Morlon, Vaibhav Srivastav, Joshua Lochner, Xuan-Son Nguyen, Colin Raffel, Leandro von Werra, and Thomas Wolf. SmolLM3: smol, multilingual, long-context reasoner. Hugging Face technical article, July 2025. URL https://huggingface.co/blog/smollm3.

L. Breiman. Stacked regressions. *Machine Learning*, 24:49–64, 1996.

G. W. Brier. Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78(1):1–3, 1950. doi: 10.1175/1520-0493(1950)078<0001:VOFEIT>2.0.CO;2.

G. Brown, J. Wyatt, and P. Tiňo. Managing diversity in regression ensembles. *Journal of Machine Learning Research*, 6:1621–1650, 2005.

Zhijun Chen, Xiaodong Lu, Jingzheng Li, Pengpeng Chen, Zhuoran Li, Kai Sun, Yuankai Luo, Qianren Mao, Ming Li, Likang Xiao, Dingqi Yang, Xiao Huang, Yikun Ban, Hailong Sun, and Philip S. Yu. Harnessing multiple large language models: A survey on LLM ensemble. *arXiv preprint arXiv:2502.18036*, 2025. URL https://arxiv.org/abs/2502.18036.

C. J. Clopper and E. S. Pearson. The use of confidence or fiducial limits illustrated in the case of the binomial. *Biometrika*, 26(4):404–413, 1934. doi: 10.1093/biomet/26.4.404.

W. G. Cochran. *Sampling Techniques*. Wiley, third edition, 1977.

A. Conneau, R. Rinott, G. Lample, A. Williams, S. R. Bowman, H. Schwenk, and V. Stoyanov. XNLI: Evaluating cross-lingual sentence representations. In *Proceedings of EMNLP*, pp. 2475–2485, 2018.

Cassio Fraga Dantas and Rémi Gribonval. Stable safe screening and structured dictionaries for faster ℓ_1 regularization. *IEEE Transactions on Signal Processing*, 67(14):3756–3769, 2019. doi: 10.1109/TSP.2019.2919404. URL https://arxiv.org/abs/1812.06635.

S. Fischer, L. Harutyunyan, M. Feurer, and B. Bischl. OpenML-CTR23: A curated tabular regression benchmarking suite. In *AutoML Conference 2023, Workshop Track*, 2023.

J. H. Friedman. Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29(5):1189–1232, 2001.

Gemma Team. Gemma 4 technical report. *arXiv preprint arXiv:2607.02770*, 2026. URL https://arxiv.org/abs/2607.02770.

Granite Team, IBM. Granite 4.1 LLMs: How they're built. Hugging Face technical article, April 2026. URL https://huggingface.co/blog/ibm-granite/granite-4-1.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, et al. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. URL https://arxiv.org/abs/2407.21783.

T. Hartvigsen, S. Gabriel, H. Palangi, M. Sap, D. Ray, and E. Kamar. ToxiGen: A large-scale machine-generated dataset for adversarial and implicit hate speech detection. arXiv:2203.09509, 2022.

W. Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30, 1963.

S. R. Howard, A. Ramdas, J. McAuliffe, and J. Sekhon. Time-uniform, nonparametric, nonasymptotic confidence sequences. *The Annals of Statistics*, 49(2):1055–1080, 2021.

F. Hutter, H. H. Hoos, and K. Leyton-Brown. Sequential model-based optimization for general algorithm configuration. In *Learning and Intelligent Optimization*, volume 6683 of *Lecture Notes in Computer Science*, pp. 507–523. Springer, 2011. doi: 10.1007/978-3-642-25566-3_40. URL https://www.cs.ubc.ca/~hutter/papers/11-LION5-SMAC.pdf.

Frank Hutter, Holger H. Hoos, Kevin Leyton-Brown, and Thomas Stützle. ParamILS: An automatic algorithm configuration framework. *Journal of Artificial Intelligence Research*, 36:267–306, 2009. doi: 10.1613/jair.2861. URL https://arxiv.org/abs/1401.3492.

T. Khot, P. Clark, M. Guerquin, P. Jansen, and A. Sabharwal. QASC: A dataset for question answering via sentence composition. In *Proceedings of AAAI*, pp. 8082–8090, 2020.

C. A. J. Klaassen and A. J. Lenstra. Why stratification may hurt, & how much. arXiv:0802.2087, 2008.

J. Kossen, S. Farquhar, Y. Gal, and T. Rainforth. Active testing: Sample-efficient model evaluation. In *Proceedings of the 38th International Conference on Machine Learning*, pp. 5753–5763, 2021.

G. Lai, Q. Xie, H. Liu, Y. Yang, and E. Hovy. RACE: Large-scale reading comprehension dataset from examinations. In *Proceedings of EMNLP*, pp. 785–794, 2017.

G. Lecué. Empirical risk minimization is optimal for the convex aggregation problem. *Bernoulli*, 19(5B):2153–2166, 2013. doi: 10.3150/12-BEJ447.

N. Li et al. The WMDP benchmark: Measuring and reducing malicious use with unlearning. In *Proceedings of the 41st International Conference on Machine Learning*, 2024.

S. Lin, J. Hilton, and O. Evans. TruthfulQA: Measuring how models mimic human falsehoods. In *Proceedings of ACL*, pp. 3214–3252, 2022.

Alexander H. Liu, Kartik Khandelwal, Sandeep Subramanian, Victor Jouault, Abhinav Rastogi, Adrien Sadé, et al. Ministral 3. *arXiv preprint arXiv:2601.08584*, 2026a. URL https://arxiv.org/abs/2601.08584.

Z. Liu, J. Zhang, C. Liu, and Y. Zhu. Active testing of large language models via approximate Neyman allocation. *arXiv preprint arXiv:2605.10075*, 2026b.

A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts. Learning word vectors for sentiment analysis. In *Proceedings of ACL-HLT*, pp. 142–150, 2011.

F. Maia Polo, L. Weber, L. Choshen, Y. Sun, G. Xu, and M. Yurochkin. tinyBenchmarks: Evaluating LLMs with fewer examples. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 34303–34326, 2024.

Oded Maron and Andrew W. Moore. Hoeffding races: Accelerating model selection search for classification and function approximation. In *Advances in Neural Information Processing Systems*, volume 6, pp. 59–66. Morgan Kaufmann, 1993. URL https://proceedings.neurips.cc/paper_files/paper/1993/hash/02a32ad2669e6fe298e607fe7cc0e1a0-Abstract.html.

N. Maslej, L. Fattorini, R. Perrault, V. Parli, A. Reuel, E. Brynjolfsson, J. Etchemendy, K. Ligett, T. Lyons, J. Manyika, J. C. Niebles, Y. Shoham, R. Wald, and J. Clark. The AI Index 2024 annual report. Technical report, AI Index Steering Committee, Institute for Human-Centered AI, Stanford University, Stanford, CA, April 2024. URL https://hai.stanford.edu/assets/files/hai_ai-index-report-2024-smaller2.pdf.

C. Mavromatis, P. Karypis, and G. Karypis. Pack of LLMs: Model fusion at test-time via perplexity optimization. In *First Conference on Language Modeling*, 2024. URL https://arxiv.org/abs/2404.11531.

Microsoft, Abdelrahman Abouelenin, Atabak Ashfaq, Adam Atkinson, Hany Awadalla, Nguyen Bach, et al. Phi-4-Mini technical report: Compact yet powerful multimodal language models via Mixture-of-LoRAs. *arXiv preprint arXiv:2503.01743*, 2025. URL https://arxiv.org/abs/2503.01743.

MiniCPM Team. MiniCPM4: Ultra-efficient LLMs on end devices. *arXiv preprint arXiv:2506.07900*, 2025. URL https://arxiv.org/abs/2506.07900.

E. Ndiaye, O. Fercoq, A. Gramfort, and J. Salmon. GAP Safe screening rules for sparse multi-task and multi-class models. In *Advances in Neural Information Processing Systems 28*, 2015.

I. Ong, A. Almahairi, V. Wu, W.-L. Chiang, T. Wu, J. E. Gonzalez, M. W. Kadous, and I. Stoica. RouteLLM: Learning to route LLMs with preference data. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=8sSqNntaMr.

M. T. Pilehvar and J. Camacho-Collados. WiC: The word-in-context dataset for evaluating context-sensitive meaning representations. In *Proceedings of NAACL-HLT*, pp. 1267–1273, 2019.

E. M. Ponti, G. Glavaš, O. Majewska, Q. Liu, I. Vulić, and A. Korhonen. XCOPA: A multilingual dataset for causal commonsense reasoning. In *Proceedings of EMNLP*, pp. 2362–2376, 2020.

Lutz Prechelt. Early stopping—but when? In Genevieve B. Orr and Klaus-Robert Müller (eds.), *Neural Networks: Tricks of the Trade*, volume 1524 of *Lecture Notes in Computer Science*, pp. 55–69. Springer, 1998. doi: 10.1007/3-540-49430-8_3. URL https://doi.org/10.1007/3-540-49430-8_3.

Project Apertus, Alejandro Hernández-Cano, Alexander Hägele, Allen Hao Huang, Angelika Romanou, Antoni-Joan Solergibert, et al. Apertus: Democratizing open and compliant LLMs for global language environments. *arXiv preprint arXiv:2509.14233*, 2025. URL https://arxiv.org/abs/2509.14233.

Qwen Team. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025. URL https://arxiv.org/abs/2505.09388.

F. Ravenda, F.-Z. Kara-Isitt, S. Swift, A. Mira, and A. Raballo. From evidence mining to meta-prediction: a gradient of methodologies for task-specific challenges in psychological assessment. In *Proceedings of the 10th Workshop on Computational Linguistics and Clinical Psychology (CLPsych 2025)*, pp. 242–248. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.clpsych-1.20. URL https://aclanthology.org/2025.clpsych-1.20/.

A. Rogers, O. Kovaleva, M. Downey, and A. Rumshisky. Getting closer to AI complete question answering: A set of prerequisite real tasks. In *Proceedings of AAAI*, 2020.

Sakana AI. Sakana Fugu technical report. arXiv:2606.21228, 2026. URL https://arxiv.org/abs/2606.21228.

O. Tafjord, M. Gardner, K. Lin, and P. Clark. QuaRTz: An open-domain dataset of qualitative relationship questions. In *Proceedings of EMNLP-IJCNLP*, pp. 5941–5946, 2019.

Team GLM, Aohan Zeng, Bin Xu, Bowen Wang, Chenhui Zhang, Da Yin, et al. ChatGLM: A family of large language models from GLM-130B to GLM-4 all tools. *arXiv preprint arXiv:2406.12793*, 2024. URL https://arxiv.org/abs/2406.12793.

Team Olmo, Allyson Ettinger, Amanda Bertsch, Bailey Kuehl, David Graham, David Heineman, et al. Olmo 3. *arXiv preprint arXiv:2512.13961*, 2025. URL https://arxiv.org/abs/2512.13961.

A. B. Tsybakov. Optimal rates of aggregation. In *Proceedings of the 16th Annual Conference on Learning Theory*, pp. 303–313, 2003.

A. Warstadt, A. Parrish, H. Liu, A. Mohananey, W. Peng, S.-F. Wang, and S. R. Bowman. BLiMP: The benchmark of linguistic minimal pairs for English. *Transactions of the Association for Computational Linguistics*, 8:377–392, 2020.

J. Wesołowski, R. Wieczorkowski, and W. Wójciak. Recursive Neyman algorithm for optimum sample allocation under box constraints on sample sizes in strata. *Survey Methodology*, 50(2): 487–511, 2024.

D. H. Wolpert. Stacked generalization. *Neural Networks*, 5(2):241–259, 1992.

D. Wood, T. Mu, A. M. Webb, H. W. J. Reeve, M. Luján, and G. Brown. A unified theory of diversity in ensemble learning. *Journal of Machine Learning Research*, 24(359):1–49, 2023.

Y. Xu, J. Lu, and J. Zhang. Bridging the gap between different vocabularies for LLM ensemble. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 7140–7152. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.naacl-long.395. URL https://aclanthology.org/2024.naacl-long.395/.

Y. Yang, Y. Zhang, C. Tar, and J. Baldridge. PAWS-X: A cross-lingual adversarial dataset for paraphrase identification. In *Proceedings of EMNLP-IJCNLP*, 2019.

W. Yu, Z. Jiang, Y. Dong, and J. Feng. ReClor: A reading comprehension dataset requiring logical reasoning. In *International Conference on Learning Representations*, 2020.

X. Zhang, J. Zhao, and Y. LeCun. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems 28*, 2015.

Jingwei Zuo, Maksim Velikanov, Ilyas Chahed, Younes Belkada, Dhia Eddine Rhayem, Guillaume Kunsch, et al. Falcon-H1: A family of hybrid-head language models redefining efficiency and performance. *arXiv preprint arXiv:2507.22448*, 2025. URL https://arxiv.org/abs/2507.22448.

A POPULATION IDENTITIES AND FINITE-SAMPLE FITTING

The population criterion concerns the best functions in the two convex hulls. Here we record the corresponding learned-weight identities and the best gain on the direct candidate segment.

A.1 LEARNED WEIGHTS AND HELD-OUT RISK

For an aggregation-training sample $D_n = \{(X_i, Y_i)\}_{i=1}^n$, let $\widehat{R}_n$ denote the empirical risk from Section 4, now viewed as a function of the random training sample. The learned incumbent and augmented predictors are

$$\widehat{f}_A \in \operatorname*{arg\,min}_{f \in \mathrm{conv}(A)} \widehat{R}_n(f), \qquad \widehat{f}_{A+b} \in \operatorname*{arg\,min}_{f \in \mathrm{conv}(A \cup \{b\})} \widehat{R}_n(f). \tag{6}$$

Conditional on D_n, the complete learned improvement is

$$\Delta_b(D_n) = R(\widehat{f}_A) - R(\widehat{f}_{A+b}). \tag{7}$$

This is a held-out or deployment quantity. If

$$\widehat{e}_A = Y - \widehat{f}_A(X), \qquad h_b = \widehat{f}_{A+b}(X) - \widehat{f}_A(X), \tag{8}$$

then expanding the two squared errors gives the complete identity

$$\Delta_b(D_n) = 2\mathbb{E}[\widehat{e}_A h_b] - \mathbb{E}[h_b^2]. \tag{9}$$

Improvement therefore rewards prediction movement aligned with the remaining error and penalizes its squared size. The identity is exact, but it is not a cheap screen because h_b is known only after fitting the augmented aggregate.

The optimized training loss cannot answer the held-out question because the augmented simplex contains the incumbent simplex and therefore cannot have higher optimized training loss.

Finite-sample transfer. The population criterion concerns the best functions in two convex hulls. After fitting their weights on a finite screening set, the corresponding held-out improvement also contains the difference in estimation error. Define

$$\mathcal{E}_A(D_n) = R(\widehat{f}_A) - R(f_A^\star), \tag{10}$$

$$\mathcal{E}_{A+b}(D_n) = R(\widehat{f}_{A+b}) - R(f_{A+b}^\star). \tag{11}$$

$$\Delta_b(D_n) = \Delta_b^\star + \mathcal{E}_A(D_n) - \mathcal{E}_{A+b}(D_n). \tag{12}$$

This follows by adding and subtracting the two population-optimal risks. Thus the alignment theorem answers whether the hull contains an improving direction, not whether a finite-sample fit will transfer. Cross-fitting, validation, or a separate generalization argument is needed for the latter. Our experiments therefore treat the labeled screening set as a finite optimization problem and score transfer on a separate held-out split.

A.2 THE BEST DIRECT SEGMENT GAIN

Using q_b from Section 3, define the gain along the direct segment $f_\lambda = (1-\lambda)f_A^\star + \lambda b$ by

$$G_b(\lambda) := R(f_A^\star) - R(f_\lambda) = 2\lambda c_b - \lambda^2 q_b, \qquad 0 \le \lambda \le 1. \tag{13}$$

Consequently,

$$\Delta_b^\star \ge G_b^\star := \max_{0 \le \lambda \le 1} G_b(\lambda) = \begin{cases} 0, & c_b \le 0, \\ c_b^2/q_b, & 0 < c_b < q_b, \\ 2c_b - q_b, & c_b \ge q_b, \end{cases} \tag{14}$$

obtained by clipping the unconstrained optimum c_b/q_b to $[0,1]$. This clipping statement applies when $q_b > 0$; if $q_b = 0$, then $d_b = 0$, $c_b = 0$, and $G_b^\star = 0$. The sign is controlled by alignment; the attainable effect also depends on how much irrelevant movement accompanies it.

B SAMPLING GUARANTEES AND SUPPORTING RESULTS

Propositions 2 and 3 apply established sampling limitations and allocation principles to candidate-alignment screening; we do not claim the underlying principles as novel.

B.1 DESIGN VARIANCE AND THE SCOPE OF POINTWISE DOMINANCE

Consider $H \geq 2$ fixed strata and a fixed allocation of B queries. Let N_h be the size of stratum h, $W_h = N_h/n$, and let n_h rows be drawn uniformly without replacement inside it. For the complete candidate contributions z_i, write

$$\mu_h = \frac{1}{N_h} \sum_{i \in h} z_i, \qquad S_h^2 = \frac{1}{N_h - 1} \sum_{i \in h} (z_i - \mu_h)^2. \tag{15}$$

The post-stratified alignment estimator $\widehat{c}_B = \sum_h W_h \overline{z}_h$ has exact conditional design variance (Cochran, 1977)

$$\mathrm{Var}(\widehat{c}_B \mid z) = \sum_h W_h^2 \left(1 - \frac{n_h}{N_h}\right) \frac{S_h^2}{n_h}. \tag{16}$$

The following counterexample instantiates the established observation that stratification can increase variance relative to simple random sampling (Klaassen & Lenstra, 2008).

Proposition 2 (No pointwise dominance over uniform sampling)**.** *Suppose every stratum contains at least two rows and receives at least one query. For any nonproportional allocation with $0 < B < n$, some unrestricted finite contribution population has larger exact design variance under that allocation than under a uniform sample of size B.*

Proof. A nonproportional allocation undersamples some proper stratum h, so $n_h/B < N_h/n$. Choose a population that varies inside this stratum and equals its stratum mean everywhere else. Its overall finite-population variance is $S^2 = (N_h - 1)S_h^2/(n-1)$. The ratio between (16) and the variance of a uniform sample is

$$\frac{W_h^2 B(n-1)}{n_h(N_h - 1)} \frac{1 - n_h/N_h}{1 - B/n} > 1, \tag{17}$$

where the first factor is greater than one because $n_h/B < W_h$ and $W_h(n-1)/(N_h - 1) > 1$, while the second is greater than one because $n_h/N_h < B/n$. Hence any fixed nonuniform allocation loses to uniform sampling for at least one placement of candidate variation. □

The counterexample places variation wherever the fixed allocation undersamples. It ranges over arbitrary contribution populations and need not satisfy a particular set of known row envelopes. It therefore does not preclude improvements for a restricted envelope class.

B.2 EXACT FINITE-POPULATION ALLOCATION AND STABILIZATION

For stratum h, set $a_h = \min_{i \in h} \ell_i$, $b_h = \max_{i \in h} u_i$, and $R_h = b_h - a_h$. The robust class allows every z_i in stratum h to range over the common outer interval $[a_h, b_h]$. The largest finite-population variance in this class is

$$V_h^{\max} = R_h^2 \frac{\lfloor N_h^2/4 \rfloor}{N_h(N_h - 1)}. \tag{18}$$

Applying capped Neyman allocation (Cochran, 1977; Wesołowski et al., 2024) to these worst-case stratum variances gives, ignoring integer rounding and the required initial coverage, the exact minimax allocation

$$n_h^{\mathrm{mm}} = \min\left\{N_h, \lambda W_h \sqrt{V_h^{\max}}\right\}, \qquad \sum_h n_h^{\mathrm{mm}} = B, \tag{19}$$

where $\lambda > 0$ is chosen to meet the budget among positive-range strata. If those strata reach their censuses first, any remaining queries can be placed arbitrarily in zero-range strata. For a continuous target share $\pi_h > 0$, define the associated concentration criterion

$$C(\pi) = \sum_h \frac{W_h^2 R_h^2}{\pi_h}. \tag{20}$$

Proposition 3 (Minimax range allocation and stabilization)**.** *Suppose a budget B assigns $n_h = B\pi_h$ queries to each stratum and samples uniformly without replacement within strata. Over the robust outer-envelope class, the worst exact design variance equals*

$$\sup_{z_i \in [a_h, b_h]} \mathrm{Var}(\widehat{c}_B \mid z) = \sum_h W_h^2 \left(1 - \frac{n_h}{N_h}\right) \frac{V_h^{\max}}{n_h}, \tag{21}$$

and (19) *minimizes this quantity over continuous allocations. For equal-sized, unsaturated strata this minimizer is $\pi^\star$, which also minimizes $C(\pi)$. Moreover, the post-stratified estimator satisfies, for every $\delta \in (0, 1)$,*

$$\Pr\left(|\widehat{c}_B - c_{b,n}| > \sqrt{\frac{\log(2/\delta)}{2B} C(\pi)}\right) \le \delta. \tag{22}$$

The stabilized target $\widetilde{\pi}$ obeys

$$C(\widetilde{\pi}) \le \frac{C(W) + C(\pi^\star)}{2} \le C(W), \qquad C(\widetilde{\pi}) \le 2C(\pi^\star). \tag{23}$$

If the target shares are realized exactly, every normalized design weight is at most $2/B$ and the Kish effective sample size is at least $B/2$.

These statements concern a fixed continuous allocation for a scalar estimate. The envelope class allows each row to take any value in its stratum's common outer interval; it can be larger than the original collection of row intervals. The implemented schedule stabilizes and discretizes this allocation to support both estimation and the downstream refit. The exact moment bounds for its realized integer counts still follow from the stated sampling design; claims requiring $n_h = B\widetilde{\pi}_h$ describe the continuous target. Appendix B.3 proves the proposition and the greedy integer scheduler's separate surrogate optimality.

B.3 Allocation proofs and integer scheduling

We first prove Proposition 3. For values in $[a_h, b_h]$, use the identity

$$S_h^2 = \frac{1}{N_h(N_h - 1)} \sum_{i<j,\ i,j \in h} (z_i - z_j)^2. \tag{24}$$

This convex function reaches its maximum over the box at an endpoint vector. If k of the N_h values use one endpoint, then

$$S_h^2 = R_h^2 \frac{k(N_h - k)}{N_h(N_h - 1)}. \tag{25}$$

The product $k(N_h - k)$ is largest at the most even split, which gives (18). Since the strata vary independently, taking this maximum in every term of (16) proves (21).

Write $A_h = W_h^2 V_h^{\max}$. Apart from the allocation-independent term $-\sum_h A_h/N_h$, minimizing the worst-case variance is equivalent to minimizing $\sum_h A_h/n_h$ subject to $\sum_h n_h = B$ and $n_h \le N_h$. The KKT conditions give $n_h \propto \sqrt{A_h} = W_h\sqrt{V_h^{\max}}$ until a stratum reaches its census, after which the remaining budget is allocated by the same rule. This is exactly (19). When all N_h are equal, the multiplier relating $\sqrt{V_h^{\max}}$ to R_h is common across strata, so the unsaturated solution is $\pi^\star$.

For the concentration statement, let $\mu_h = N_h^{-1}\sum_{i \in h} z_i$ be the complete mean in stratum h. A uniform sample without replacement has $\mathbb{E}[\overline{z}_h] = \mu_h$, so $\mathbb{E}[\widehat{c}_B] = \sum_h W_h \mu_h = c_{b,n}$. Hoeffding's comparison theorem for sampling without replacement gives the same bounded-variable moment bound as independent sampling with replacement (Hoeffding, 1963). Samples are independent across strata, hence

$$\Pr(|\widehat{c}_B - c_{b,n}| > t) \le 2\exp\left\{-\frac{2t^2}{\sum_h W_h^2 R_h^2/n_h}\right\}. \tag{26}$$

Substituting $n_h = B\pi_h$ proves (22).

For any allocation π on the simplex, Cauchy–Schwarz gives

$$C(\pi) = \sum_h \frac{(W_h R_h)^2}{\pi_h} \geq \left(\sum_h W_h R_h \right)^2, \tag{27}$$

with equality at $\pi = \pi^\star$. Terms with $W_h R_h = 0$ contribute zero. The map $\pi \mapsto C(\pi)$ is convex, so for $\widetilde{\pi} = (W + \pi^\star)/2$,

$$C(\widetilde{\pi}) \leq \frac{C(W) + C(\pi^\star)}{2} \leq C(W). \tag{28}$$

Also $\widetilde{\pi}_h \geq \pi^\star_h/2$, which gives $C(\widetilde{\pi}) \leq 2C(\pi^\star)$. If $R_{\text{all}} = \max_i u_i - \min_i \ell_i$, then $C(W) = \sum_h W_h R_h^2 \leq R_{\text{all}}^2$, recovering a bound no larger than the global uniform-range criterion.

When $n_h = B\widetilde{\pi}_h$ exactly, one sampled row in stratum h has normalized design weight

$$\omega_i = \frac{W_h}{n_h} = \frac{W_h}{B\widetilde{\pi}_h} \leq \frac{2}{B}. \tag{29}$$

The weights sum to one, so $\sum_i \omega_i^2 \leq (2/B) \sum_i \omega_i = 2/B$ and the Kish effective sample size is at least $B/2$. This completes the proof.

The implementation must use integer counts and one nested path for every budget. For a fixed target $\widetilde{\pi}$, define

$$G_{\widetilde{\pi}}(n_1, \ldots, n_H) = \sum_h \frac{\widetilde{\pi}_h^2}{n_h}, \qquad 1 \leq n_h \leq N_h. \tag{30}$$

Adding one query to stratum h decreases $G_{\widetilde{\pi}}$ by exactly $\widetilde{\pi}_h^2/[n_h(n_h+1)]$. These marginal decreases are nonincreasing within each stratum. Selecting the largest available decrease at each step therefore minimizes $G_{\widetilde{\pi}}$ for every integer budget, subject to the coverage and population constraints. This is the greedy rule used by the nested scheduler. The continuous factor-two and weight statements describe its target; the variance diagnostic below uses the realized integer counts instead of treating them as exactly $B\widetilde{\pi}_h$.

B.4 Exact design variance and a limitation

Equation (16) is available only after the candidate outputs are known, so we use it as a post-hoc mechanism diagnostic. Replacing the exact maximum in (18) by the simpler upper bound $N_h R_h^2/[4(N_h - 1)]$ gives

$$\mathrm{Var}(\widehat{c}_B \mid z) \leq \sum_h W_h^2 \left(1 - \frac{n_h}{N_h}\right) \frac{N_h R_h^2}{4(N_h - 1) n_h}. \tag{31}$$

Both the exact variance and this bound vanish at a census. Proposition 2 says the realized ratio cannot favor the residual-aware allocation for every finite contribution population. The diagnostic below asks how often it does so on the frozen panels and how closely the robust bound tracks the realized variance.

Figure 3 reports the full budget curves. At 64 calls, loss-based allocation reduces exact variance by 61.4% in regression, 62.4% on unseen LLM tasks, and 64.0% on unseen LLM models. Alignment-envelope allocation reduces it by 60.9%, 58.2%, and 60.0%. Every one of the 35 regression datasets and 15 LLM task families in each panel has lower aggregated exact variance than uniform sampling. In contrast, population-proportional allocation inside the same residual strata reduces pooled exact variance by only 1.0%, 8.0%, and 7.7%. The range-bound ratio is more optimistic than the realized ratio, especially for proportional strata; it supports the allocation criterion but does not predict its effect size.

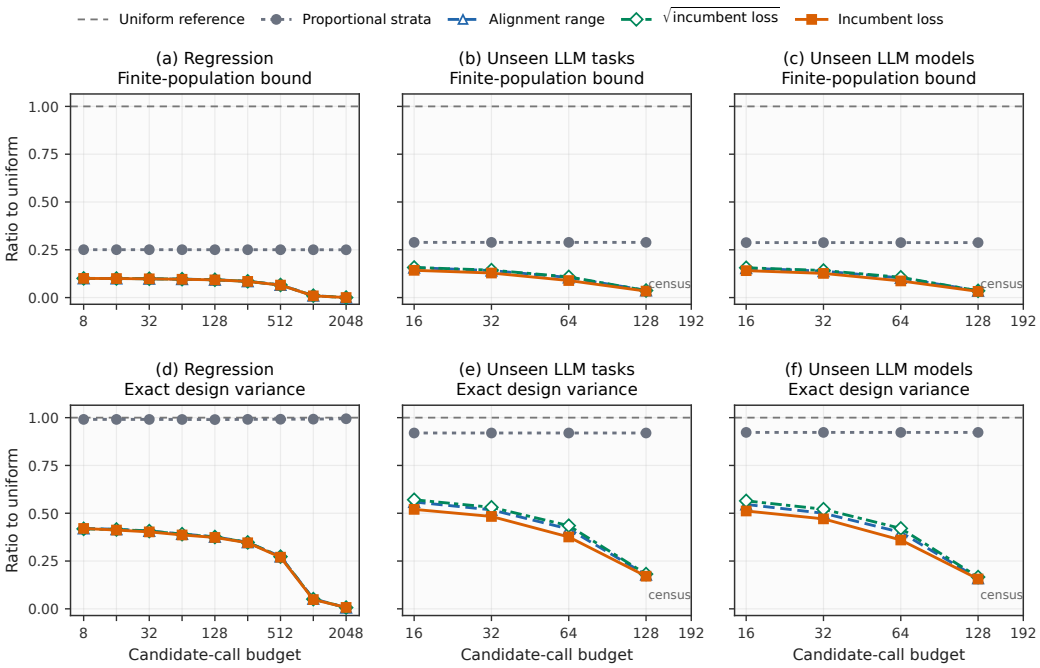

Figure 3: **Mechanism diagnostic for residual-aware sampling.** Columns show regression, unseen LLM task families, and unseen LLM model families. Top: the finite-population range bound in (31), divided by the corresponding uniform bound. Bottom: exact design variance from (16), divided by uniform variance. Values below one favor the residual-aware schedule. QP stopping and convex refits are disabled. The 192-row LLM census is omitted because every exact variance and finite-population bound is zero.

Table 2: Exact design variance of the alignment estimator at 64 calls, reported as the ratio of summed case variances to the corresponding sum under uniform sampling. Lower is better. "Proportional strata" uses alignment-width strata but allocates queries by stratum population alone. The square-root-loss control uses scheduling pseudo-intervals $[0, \sqrt{\eta_i}]$. The last column applies to all four schedules. Full budget curves and the range-bound diagnostic appear in Figure 3 in the appendix.

Panel	Proportional strata	Alignment	$\sqrt{\text{loss}}$	Loss	Tasks below uniform
Regression	0.990	0.391	0.390	**0.386**	35/35
LLMs: unseen tasks	0.920	0.418	0.435	**0.376**	15/15
LLMs: unseen models	0.923	0.400	0.421	**0.360**	15/15

B.5 Proof of the loss–headroom relation

Proof. The scalar identity follows immediately from $u_i - \ell_i = |r_{A,n,i}|(U_i - L_i)$. For Brier loss, suppress the row index, write $p = p^\star_A$, let t be the true class, and set

$$x = 1 - p_t, \qquad m = \max_{k \neq t} p_k, \qquad s_2 = \sum_{k \neq t} p_k^2. \tag{32}$$

The largest alignment contribution is attained by the perfect prediction e_t, so

$$u = \|e_t - p\|_2^2 = x^2 + s_2 = \eta. \tag{33}$$

The largest and smallest entries of the residual vector $e_t - p$ are x and $-m$, respectively, so the alignment interval has width $w = x + m$. Since $m^2 \leq s_2 \leq xm$,

$$\frac{w^2}{2} \leq x^2 + s_2 \leq w^2, \tag{34}$$

which proves $\sqrt{\eta_i} \leq w_i \leq \sqrt{2\eta_i}$.. With two classes, $m = x$ and $s_2 = x^2$, so equality gives $\eta = w^2/2$. □

C Supporting completion certificates

C.1 Why the certificate is one-sided

A zero-error early-acceptance rule based on the same partial information would instead require

$$L_q := \min_{v \in \mathcal{B}_q} \Delta_n(v) = R_n(f^\star_{A,n}) - \max_{v \in \mathcal{B}_q} \min_{g \in \text{conv}(A \cup \{v\})} R_n(g) \geq \tau_n. \tag{35}$$

Unlike U_q, this is a max–min problem: the completion is adversarial, while the ensemble may be re-optimized for that completion. The perspective argument above therefore does not produce the same convex QP by reversing the inequality. Interchanging the optimizations gives a safe sufficient condition, but requires one fixed ensemble weighting to attain the target for every compatible completion and can be highly conservative for broad completion sets. We therefore certify early rejection only and reserve positive decisions for the budget refit or evaluation of the full screening set.

C.2 Relation to alignment-only screening

Proposition 4 (Necessary alignment for material gain on the screening set)**.** *For squared loss and simplex aggregation,*

$$\Delta^\star_{b,n} \leq 2\,[c_{b,n}]_+\,. \tag{36}$$

Consequently, $\Delta^\star_{b,n} \geq \tau_n$ *requires* $c_{b,n} \geq \tau_n/2$.

Appendix C.3 proves the proposition and derives its closed-form partial-query rejection rule. Positive alignment is necessary but does not guarantee a one-percent gain. The exact QP subsumes this scalar screen when both use the same completion set. Indeed, write $c_n(v) = \left\langle r_{A,n}, v - f^\star_{A,n} \right\rangle_n$. Proposition 4 gives

$$U_q \leq 2 \max_{v \in \mathcal{B}_q} [c_n(v)]_+\,, \tag{37}$$

which is the optimistic alignment-completion bound. Different completion sets need not preserve this ordering.

C.3 ALIGNMENT-COMPLETION REJECTION

The alignment inequality yields a closed-form rejection certificate even without solving the QP. It is computationally cheap, but it discards the quadratic curvature used by the exact optimistic completion. We retain it as a theoretical comparator rather than as part of the final policy.

Proof of Proposition 4. Every augmented prediction has the form $g = (1-\lambda)a + \lambda b$, with $a \in \mathrm{conv}(A)$ and $\lambda \in [0,1]$. Projection optimality gives $\langle r_{A,n}, a - f^\star_{A,n} \rangle_n \le 0$. For $h = g - f^\star_{A,n}$,

$$R_n(f^\star_{A,n}) - R_n(g) = 2\langle r_{A,n}, h\rangle_n - \|h\|_n^2 \le 2\lambda c_{b,n} \le 2\,[c_{b,n}]_+ . \tag{38}$$

Applying the inequality to $g = f^\star_{A+b,n}$ proves (36). □

Corollary 2 (Alignment-completion rejection)**.** *Let $\mathcal{I}_q$ be the queried rows after q calls. With z_i and u_i defined in Section 4.2, set*

$$\overline{c}_q = \frac{1}{n}\left(\sum_{i\in\mathcal{I}_q} z_i + \sum_{i\notin\mathcal{I}_q} u_i\right). \tag{39}$$

If $2\,[\overline{c}_q]_+ < \tau_n$, then the candidate cannot achieve the material gain τ_n on the screening set.

Proof. Every unseen z_i is at most u_i, so $c_{b,n} \le \overline{c}_q$. Proposition 4 gives

$$\Delta^\star_{b,n} \le 2\,[c_{b,n}]_+ \le 2\,[\overline{c}_q]_+ . \tag{40}$$

Thus $2\,[\overline{c}_q]_+ < \tau_n$ certifies failure to meet the required gain on the screening set. The same argument applies to the vector inner products used for multiclass Brier loss. □

C.4 WEAK DUALITY AND THE NUMERICAL WITNESS

The following expressions use coordinate-average risk. In scalar regression $d = n$, so this equals MSE on the screening set. For flattened Brier coordinates multiply risks and gain bounds by d/n to recover loss per question.

For any $s \in \mathbb{R}^d$, write s_{ik} for its entry corresponding to class k of example i. For scalar predictions, take $K_i = 1$ and identify $s_{i1} = s_i$, and likewise for candidate predictions and bounds. Define the support function of the completion box by

$$h_q(s) = \sum_{i\in\mathcal{I}_q}\sum_{k=1}^{K_i} s_{ik}b_{ik} + \sum_{i\notin\mathcal{I}_q}\sum_{k=1}^{K_i} \max\{s_{ik}L_{ik}, s_{ik}U_{ik}\}, \tag{41}$$

and the explicit weak-dual value

$$D_q(s) = 2s^\top y - d\|s\|_2^2 - 2\max\left\{\max_{1\le j\le m} a_j^\top s,\ h_q(s)\right\}. \tag{42}$$

Proposition 5 (Solver-independent weak-dual certificate)**.** *For every $s \in \mathbb{R}^d$,*

$$D_q(s) \le \rho_q^{\mathrm{box}} \le \rho_q^{\mathrm{true}}, \tag{43}$$

where $\rho_q^{\mathrm{true}} = \min_{g\in\mathrm{conv}(A\cup\{b\})} \|y-g\|_2^2/d$ is augmented coordinate-average risk for the fixed, complete candidate. Consequently,

$$\Delta^\star_{b,n} \le R_n(f^\star_{A,n}) - \frac{d}{n} D_q(s). \tag{44}$$

The first inequality remains valid for any finite value of s; optimizer convergence is not required.

Proof. For every prediction vector g and every s,

$$\frac{1}{d}\|y-g\|_2^2 \ge 2s^\top(y-g) - d\|s\|_2^2. \tag{45}$$

Minimizing the left side over the convex hull of the incumbents and $\mathcal{B}_q$, then maximizing $s^\top g$ on the right, gives $D_q(s) \le \rho_q^{\mathrm{box}}$. Fixing the completion to the actual candidate can only increase the optimum, which gives the second inequality. Converting the lower bound to empirical-risk units and subtracting it from incumbent empirical risk proves (44). □

C.5 MULTICLASS COMPLETION

For Brier loss, let $b_{ik} = (p_{b,i})_k$ denote the candidate's predicted probability for class k of example i, with $k = 1, \ldots, K_i$. Concatenating the probability vectors gives $d = \sum_{i=1}^{n} K_i$ entries, indexed by pairs (i, k). Querying example i reveals all its K_i probabilities.

The completion set $\mathcal{B}_q$ fixes $v_{ik} = b_{ik}$ for $i \in \mathcal{I}_q$ and allows $L_{ik} \le v_{ik} \le U_{ik}$ for $i \notin \mathcal{I}_q$, for every valid class k. The corresponding QP constraints become

$$\begin{aligned} t_{ik} &= \beta b_{ik} && (i \in \mathcal{I}_q), \\ \beta L_{ik} \le t_{ik} &\le \beta U_{ik} && (i \notin \mathcal{I}_q), \end{aligned}$$

for every $k = 1, \ldots, K_i$. Here, y concatenates the one-hot targets, and each column of $\mathbf{A}$ concatenates an incumbent's probability vectors.

The factor d/n in (55) converts coordinate-average loss to Brier loss per question, giving the exact compatible gain

$$R_n(f^{\star}_{A,n}) - \frac{d}{n}\rho_q^{\mathrm{box}}. \tag{46}$$

Enforcing the known probability simplex would add only the linear constraints

$$t_{ik} \ge 0, \qquad \sum_{k=1}^{K_i} t_{ik} = \beta, \tag{47}$$

preserving exactness for that set.

The experiments use the larger coordinate box $[0, 1]^{K_i}$, with $L_{ik} = 0$ and $U_{ik} = 1$. It contains the true candidate and also allows extra completions that can delay rejection. Completeness for this box therefore does not imply completeness for all the available simplex information.

D EXPERIMENTAL SETUP AND REPRODUCIBILITY

D.1 REGRESSION PANEL

We use the 35 regression tasks in OpenML suite 353 (OpenML-CTR23) (Fischer et al., 2023). Each task is capped at 12,000 rows and split into 60% model training, 25% screening data, and 15% held-out data. Targets are standardized from the training split, predictions are clipped to $[-4, 4]$, and screening set sizes range from 129 to 3,000 rows.

For each task and ten model seeds, we train three variants from ridge regression, random forest, histogram gradient boosting, and a multilayer perceptron. A case holds out one family: its third, highest-capacity recipe is the candidate, while the incumbents are the first two variants from each of the other three families. The panel therefore contains

$$35 \text{ tasks} \times 10 \text{ seeds} \times 4 \text{ families} = 1{,}400 \text{ candidate cases}. \tag{48}$$

The primary target is the decision from the full screening set in Equation (2). For the supplementary *held-out useful* label, we fit simplex weights on the full screening set for A and $A+b$, freeze both fits, and evaluate their held-out mean squared errors. For positive incumbent held-out risk, a candidate is useful when

$$\frac{R_{A,\mathrm{test}} - R_{A+b,\mathrm{test}}}{R_{A,\mathrm{test}}} \ge 0.01. \tag{49}$$

There are 638 useful cases, or 45.57%. Policies never receive held-out labels, predictions, or risks.

The nominal budgets are

$$B \in \{8, 16, 32, 64, 128, 256, 512, 1024, 2048\}. \tag{50}$$

Budgets through 128 fit every screening set. A larger budget is truncated when the screening set is smaller; the budget-trajectory figures use nominal caps on the horizontal axis.

D.2 LLM PANELS

The original panel contains the first eight models in Table 1. In the task-family experiment, each model becomes the candidate while the other seven are incumbents. In the model-family experiment, the original eight remain fixed as set A and each of the four new-family models is added separately. All evaluated artifacts are pinned Q4 GGUF files. New-candidate predictions, caches, and run outputs use a separate namespace and cannot enter set A or one another's fits.

For background on the model families and releases listed in Table 1, see the reports and technical articles for Qwen3 (Qwen Team, 2025), Llama 3 (Grattafiori et al., 2024), Gemma 4 (Gemma Team, 2026), Ministral 3 (Liu et al., 2026a), Phi-4 Mini (Microsoft et al., 2025), Granite 4.1 (Granite Team, IBM, 2026), OLMo 3 (Team Olmo et al., 2025), Falcon-H1 (Zuo et al., 2025), SmolLM3 (Bakouch et al., 2025), MiniCPM4 (MiniCPM Team, 2025), GLM-4 (Team GLM et al., 2024), and Apertus (Project Apertus et al., 2025).

The 15 task families are RACE-High, BLiMP adjunct-island, ToxiGen, WiC, PAWS-X English, XNLI English, WMDP-Cyber, XCOPA Italian, TruthfulQA MC1, QASC, QuaRTz, QuAIL, AG News, IMDb, and ReClor (Lai et al., 2017; Warstadt et al., 2020; Hartvigsen et al., 2022; Pilehvar & Camacho-Collados, 2019; Yang et al., 2019; Conneau et al., 2018; Li et al., 2024; Ponti et al., 2020; Lin et al., 2022; Khot et al., 2020; Tafjord et al., 2019; Rogers et al., 2020; Zhang et al., 2015; Maas et al., 2011; Yu et al., 2020). Passages, premises, and reused supporting facts remain intact across the screening set and held-out split. Every task supplies a screening set of 192 rows. The primary held-out set has 320 rows for 12 tasks, 318 for XNLI, and 192 for XCOPA and ReClor.

A constrained next-token query returns a normalized probability vector over the rendered choices. A tokenizer and template census checked that every request fits the 4,096-token context. We fit incumbent and augmented convex weights on the 192-row screening set and threshold the resulting gain for the primary decision. For the supplementary held-out usefulness label, we freeze those weights and require at least a one-percent reduction in held-out Brier loss. No held-out candidate probability enters a query path, fallback fit, or certificate.

The task-family gate contains $15 \times 8 = 120$ cases; the model-family gate contains $15 \times 4 = 60$. Each case is replayed on ten deterministic nested paths at caps $16, 32, 64, 128,$ and 192. The resulting 1,200 and 600 path decisions are repeated measurements rather than independent cases.

Table 3: Supplementary model metrics for the introduction's accuracy-oracle comparison in Table 1. Standalone Brier is the task-macro mean summed multiclass loss per question. Selected-task counts use strictly positive held-out accuracy gain; these counts determine the oracle call savings in Table 1.

Model	Family	Standalone Brier ↓	Accuracy-oracle selected tasks / 15
Qwen3 Instruct 2507	Qwen	0.4091	7
Llama 3.2 Instruct	Llama	0.4976	2
Gemma 4 E2B IT	Gemma	0.4804	4
Ministral 3 Instruct 2512	Mistral	0.4081	4
Phi-4 Mini Instruct	Phi	0.4390	2
Granite 4.1	Granite	0.5074	5
OLMo 3 Instruct	OLMo	0.4801	5
Falcon-H1 Instruct	Falcon	0.3391	9
SmolLM3	SmolLM	0.5031	4
MiniCPM 4.1	MiniCPM	0.4379	5
GLM-4 0414	GLM	0.3349	8
Apertus Instruct 2509	Apertus	0.5862	1

The original-eight aggregate reaches 81.11% held-out accuracy, compared with 78.59% for single-model selection on the screening set, and lowers macro Brier from 0.3259 to 0.2662. The single-model baseline chooses one original model per task by Brier loss on the screening set. The aggregate fits task-specific convex weights on the screening set before held-out evaluation.

Table 1 reports task-macro standalone accuracy and the numbers of tasks helped, hurt, or unchanged by adding each candidate. Effects compare held-out top-1 accuracy with and without the candidate, after fitting both sets of convex weights by Brier loss on the screening set. The parenthesized changes average only over the tasks in their respective helped or hurt group; unchanged means equal accuracy, not identical probabilities. The three counts sum to 15. Table 3 gives full model names, supporting Brier scores, and oracle task counts. The accuracy oracle queries the candidate on all 192 screening rows of each helped task and skips hurt and unchanged tasks. Its call savings are $100(15 - n_{\mathrm{helped}})/15$, with incumbent predictions already cached. This assumes knowledge of held-out outcomes before querying and measures neither an implementable screening policy nor total deployment savings.

Table 4 gives the parallel analysis for the supplementary held-out Brier usefulness label. An original model is compared with the other seven, whereas a new-family candidate is compared with the fixed original eight. No joint 12-model ensemble is fitted. Names, family labels, and nominal size classes follow the frozen repository metadata; size is not GGUF file size. E2B retains Google's effective-parameter size designation for Gemma 4.

Table 4: Descriptive effects under the supplementary held-out Brier usefulness label. A task is Brier-useful when the augmented aggregate fitted on the full screening set reduces held-out Brier loss by at least one percent relative to the incumbent. Entries are task-macro means of the signed relative Brier reduction; positive is better. "Useful only" assigns zero change to other tasks, so both policies retain the same 15-task denominator. Within the two reduction columns, bold marks the larger result in each row. Always adding can win when excluded tasks have a positive net contribution, including gains below the one-percent threshold. This is a held-out-oracle diagnostic, not an operational routing policy.

Model	Brier-useful tasks / 15	Relative Brier reduction (%) always add	Relative Brier reduction (%) useful only
Original panel			
Qwen3 Instruct 2507	8	+1.959	**+2.554**
Llama 3.2 Instruct	3	−0.052	**+0.308**
Gemma 4 E2B IT	2	+0.005	**+0.218**
Ministral 3 Instruct 2512	4	**+1.261**	+1.128
Phi-4 Mini Instruct	1	+0.159	**+0.245**
Granite 4.1	3	+0.579	**+0.780**
OLMo 3 Instruct	3	+0.963	**+1.334**
Falcon-H1 Instruct	12	+9.570	**+10.044**
New-family candidates (held out from development)			
SmolLM3	0	−0.013	**+0.000**
MiniCPM 4.1	5	+1.576	**+1.581**
GLM-4 0414	11	**+7.604**	+7.555
Apertus Instruct 2509	4	+0.658	**+0.728**
Across-candidate mean	—	+2.022	**+2.206**

D.3 Matched protocols and uncertainty

The fixed-budget sampling comparison has no early stopping. Uniform sampling uses a nested uniform prefix. The two residual-aware methods form four strata from incumbent loss or the alignment envelope, apply the same stabilized allocation, and fit the same post-stratified full-simplex objective. Every method reaches the cap and predicts useful when fitted relative gain is at least one percent.

The stopping comparison fixes the alignment-stratified path, checkpoints, post-stratified fit, and one-percent fallback. No stopping always reaches the cap. Two-negative stability rejects after two consecutive distinct positive-query checkpoints predict useless and never accepts early. The alignment-completion rule uses Corollary 2; the coordinate-box QP uses Proposition 5. Unresolved cases reach the shared cap and fallback.

The conditional-scenario comparison keeps that path and fallback fixed. It is inactive through 64 queries and considers $\delta \in \{0.01, 0.03, 0.05, 0.10, 0.20\}$. Each active checkpoint requests up to 512 complete posterior draws. A rejected checkpoint evaluates all 512; the LLM replay can short-circuit a non-rejecting checkpoint after exceeding the active material-count cutoff. The regression replay reuses random numbers across checkpoints and thresholds on a path. The LLM replay uses a checkpoint-specific seed and reuses draws only across thresholds within that checkpoint. A scenario is material when its exactly reoptimized relative gain is at least one percent. Rejection requires the pointwise one-sided 99% Clopper-Pearson upper bound (Clopper & Pearson, 1934) on the material-scenario probability to be below δ. The plotted points are paired replay means on the same ten paths per case; δ was swept rather than chosen on an independent validation panel.

For stopping rule s at cap B, the primary metrics are $100A_{s,B}$, where $A_{s,B}$ is the fraction of decisions matching reaggregation on the full screening set, and call saving relative to the full screening set

$$S_{s,B}^{\text{full}} = 100\left(1 - \frac{Q_{s,B}}{\bar{n}}\right). \tag{51}$$

Here $Q_{s,B}$ is mean calls and $\bar{n}$ is mean screening set size under the panel weighting below. Regression has $\bar{n} = 1{,}712.23$ rows, while both LLM panels have $\bar{n} = 192$. For no stopping, $Q_{\text{no stop},B}$ is the panel mean of $\min(B, n)$. The ratio of means counts the fraction of total candidate calls avoided under that weighting; it is not the mean of per-case saving fractions.

Supplementary stopping controls and the posterior threshold sweep instead isolate early stopping at a fixed cap. They report matched-cap call saving

$$100\frac{Q_{\text{no stop},B} - Q_{s,B}}{Q_{\text{no stop},B}}, \tag{52}$$

and change in screening accuracy

$$100\left(A_{s,B} - A_{\text{no stop},B}\right), \tag{53}$$

These supplementary quantities use capped no stopping as their baseline; they are distinct from the absolute metrics in Figures 2 (a) and 2 (b).

All panels use ten nested paths per case and 5,000 paired hierarchical bootstrap draws. Regression estimates average paths within cases, then seeds and candidate families within datasets, and finally weight the 35 datasets equally. Its bootstrap resamples datasets, seeds within datasets, and paths within seeds while retaining the four candidates jointly. The LLM bootstrap resamples task families and paths within family while retaining the fixed candidate panel. Draws are shared across methods and axes, and ratios are computed inside each draw. The independent LLM task dimension therefore has 15 clusters. The four new candidates are a fixed purposive set, so their intervals do not support a population-level model-family claim.

An always-reject rule gives zero-query held-out usefulness-label accuracies of 54.43% in regression, 70.00% for unseen LLM tasks, and 66.67% for unseen LLM models. It is omitted from the stopping comparison because it is a degenerate prevalence baseline that never screens a candidate. Reaggregation on the full screening set is a maximum-information reference, not perfect held-out ground truth. The regression sampling and stopping controls are exploratory replays on an already studied panel.

D.4 Certificate implementation

Conservative numerical evaluation. To certify rejection numerically, we need a lower bound on the optimistic risk. A solver's objective value alone does not provide that bound. The implementation evaluates an explicit weak-dual lower bound $D_q(s)$ at finite dual-feasible witness vectors s. It satisfies

$$D_q(s) \le \rho_q^{\text{box}} \tag{54}$$

in exact arithmetic, even without solver convergence (Proposition 5).

We subtract a small numerical cushion and retain the largest accepted lower bound along the path, denoted $\underline{\rho}_q^{\text{QP}}$. To express this bound as loss per screening example, we use

$$\underline{R}_q^{\text{QP}} = \frac{d}{n}\,\underline{\rho}_q^{\text{QP}}. \tag{55}$$

Here, n counts screening examples and d counts prediction coordinates; for scalar regression, $d = n$. Underlines and overlines denote lower and upper bounds.

Subtracting this lower bound from the incumbent loss gives an upper bound on the improvement achievable by adding the candidate:

$$\overline{\Delta}_q^{\mathrm{QP}} = \max\{0,\ R_n(f^\star_{A,n}) - \underline{R}_q^{\mathrm{QP}}\}\,. \tag{56}$$

The final policy is to reject when $\overline{\Delta}_q^{\mathrm{QP}} < \tau_n$. A solver failure returns no new bound.

Because approximate witnesses may give weaker bounds, the implementation can reject later than the exact optimum and does not inherit its completeness. The floating-point cushion guards against numerical error but is not a formal rounding-error proof. The implementation details below specify the tolerances and failure handling.

For Brier loss, each query reveals a full probability vector; the LLM implementation bounds unobserved probabilities individually in $[0, 1]$, preserving safety but potentially delaying rejection (Appendix C.5).

Solver and dual witnesses. The implementation eliminates t entry by entry. In scalar regression, for an unqueried example it clips $y_i - (A\alpha)_i$ to $[\beta L_i, \beta U_i]$. For multiclass predictions, the same clipping applies independently to each class entry (i, k) of an unqueried example, using $[\beta L_{ik}, \beta U_{ik}]$; queried examples have all entries fixed at $t_{ik} = \beta b_{ik}$. Thus SLSQP optimizes only the $m + 1$ simplex weights with an analytic gradient. From both the starting point and the projected solver output, it forms a residual, divides it by d, and uses the result as a witness in (42). If these witnesses form the set $\mathcal{S}_q$ at checkpoint q, the returned values are

$$\ell_q = \max\left\{0,\ \max_{s\in\mathcal{S}_q} D_q(s) - \varepsilon_{\mathrm{num}}\right\}, \qquad \underline{\rho}_q^{\mathrm{QP}} = \max_{q'\le q} \ell_{q'}. \tag{57}$$

The second maximum carries lower-bound values from earlier checkpoints; it does not reevaluate their witnesses under the new completion box. This remains valid because querying an example fixes its entire prediction vector, shrinks $\mathcal{B}_q$, and makes ρ_q^{box} nondecreasing.

Numerical tolerances and failure handling. We use $\varepsilon_{\mathrm{num}} = 10^{-12} + 10^{-10}S$, where S is the largest of one, the mean squared outcomes, the mean squared incumbent predictions, and the squared prediction bounds. The subtraction is a floating-point guard, not an interval-arithmetic proof. SLSQP uses a 10^{-12} objective tolerance and at most 300 iterations. A solver exception returns no new lower bound. Nonfinite or numerically inconsistent witness values are discarded, while invalid inputs stop the run. A nonconverged projected iterate may still yield a valid weak dual value because Proposition 5 holds for every finite witness. A later solver failure leaves any earlier valid path-level bound in place.

Evaluation schedule and computational cost. The policy checks the certificate at zero queries, every 32 queries, and each budget before the census. The census uses exact reaggregation on the screening set. Eliminating t leaves $m + 1$ optimization variables. One objective, gradient, or dual evaluation costs $O(dm)$. With at most $I = 300$ SLSQP iterations, a conservative per-checkpoint bound is $O(I(dm + m^3))$ time and $O(dm + m^2)$ storage. In the Brier experiments, d is the number of valid class coordinates rather than the number of questions.

D.5 Stopping-policy overhead and inference-time reference

Candidate-call savings translate into wall-clock savings only when the avoided inference costs more than the additional stopping-policy work. For rule s, we summarize this threshold by the break-even candidate time

$$t_{\mathrm{BE}}(s) = \frac{T_s}{\sum_p (Q_{\text{no stop},p} - Q_{s,p})}, \tag{58}$$

where T_s is the aggregate rule-specific runtime, measured or estimated as described below, over the evaluated paths. The denominator counts calls avoided relative to matched capped no stopping, isolating the additional savings due to the stopping rule. If marginal candidate inference exceeds

$t_{\mathrm{BE}}(s)$, its estimated time saved exceeds the estimated policy overhead. Paths with no saving remain in the runtime numerator, so the calculation includes unsuccessful stopping checks.

The manuscript record places both original timing sources on a MacBook Pro with an Apple M5 Pro processor (18 CPU cores) and 64 GB of unified memory. It reports Python 3.12, NumPy 2.5.2, SciPy 1.18.1, and one-thread BLAS/OpenMP for the safe-QP benchmark. No compact raw safe-QP timing manifest or original inference logs were retained. Those timing values and protocol metadata are therefore historical manuscript provenance and cannot be independently reproduced from the repository; only the posterior-overlay timings below are now retained compactly. The original record reports five warmed repetitions over the exact frozen LLM paths and the median total time. The timed work includes coordinate construction, the zero-query certificate, warm starts, and every SLSQP certificate solve reached before stopping. It excludes alignment-path construction and the final post-stratified refit, which are shared with the matched no-stop comparison. Every reproduced QP stop matched the frozen path record.

Table 5: Historical QP overhead and panel-level break-even inference time at the 192-call cap. Times are totals over all paths, not per-path times.

Panel	Paths	Calls avoided	QP time, median [range] (s)	t_{BE} (ms/call)
Unseen LLM tasks	1,200	53,056	$4.590\ [4.574, 4.598]$	0.087
Unseen LLM models	600	24,672	$2.383\ [2.379, 2.385]$	0.097

Collection notes place the posterior timing audit on the same hardware but in the current locked environment: Python 3.12.13, NumPy 2.5.2, SciPy 1.18.0, pandas 2.3.3, and Matplotlib 3.11.1. These versions were inspected at evidence collection rather than embedded in the temporary run manifests, and their BLAS/OpenMP limits were not recorded. We timed the $\delta = 0.05$ overlay for Qwen3 4B on its 15 cases and ten paths per case, using one process worker, 512 scenarios, one warm-up, and five measured repetitions. The marginal-box overlay takes a median 1.0881 s $[1.0778, 1.0957]$ and performs 600 posterior-bound QP solves. The conditional overlay takes 11.3183 s $[11.0882, 11.6756]$ and performs 30,067 exact scenario refits per repetition. These overlays read the stored safe-QP trace. Adding the approximately 0.6406 s implied by the rounded 0.286 ms/call safe-QP value and 2,240 avoided calls gives the implied integrated estimates in Table 6; shared work was not timed as one process.

Table 6 summarizes the Qwen3 4B comparison. A safe checkpoint solves one optimistic QP. A marginal posterior box adds one posterior-bound QP per threshold and checkpoint. Conditional scenarios require an exact augmented-ensemble refit per retained draw, with early abort when rejection becomes impossible. The marginal box changes no Qwen stopping outcome, whereas scenarios avoid 3,840 calls, 71.4% more than the safe QP. Their approximate 12.0 s policy total is about 80 ms per path and 3.7% of the estimated avoided inference time. Inference depends on hardware, batching, model size, quantization, prompt and output length, and inference-time reasoning; this calculation is an approximate implementation estimate.

Table 6: Candidate-specific stopping-policy overhead for Qwen3 4B at the 192-call cap. Break-even time divides policy computation by calls avoided; the last column divides the 83.45 ms mean inference request by that value. Posterior totals combine replay time with the safe-QP time implied by its previously measured break-even value and are rounded because shared work was not timed as one integrated implementation. Replay timing includes input loading and validation but excludes artifact writing and plotting. These are overhead margins, not end-to-end speedups.

Policy	Calls avoided	Est. policy time (s)	t_{BE} (ms/call)	Inference / overhead
Safe optimistic QP	2,240	≈ 0.641	0.286	$292\times$
Marginal posterior box, $\delta = .05$	2,240	≈ 1.7	≈ 0.77	$\approx 108\times$
Conditional scenarios, $\delta = .05$	3,840	≈ 12.0	≈ 3.11	$\approx 26.8\times$

For batch-throughput context only, the six-worker, all-δ replays take 263.6 s for 14,000 regression paths and 54.8 s for 1,800 LLM paths. Those runs amortize setup across five thresholds and use early abort, so they are not single-policy online latency measurements.

For the inference reference, the frozen runs used `llama.cpp` 0.2.0, build 10566 (commit `bb4caa754`), six inference threads, one slot, and a 4,096-token context. Each request produced a constrained one-token score. We read `slot print_timing: total time` from the server logs; this excludes model loading, client work, and network latency. After removing one warm-up request from each original-panel log and 11 diagnostic or preflight requests from each new-model log, every model contributes 7,422 production requests. The logs preserve the original sequential request order and its KV/prefix-cache behavior.

Table 7 pairs these request times with a break-even value computed by restricting Equation 58 to each candidate. The ratio uses the corpus-wide mean request time as the per-call reference; it does not claim that the evaluation is that many times faster end to end.

Table 7: Historical frozen-run one-token inference timing and candidate-specific QP break-even time on the same machine. The Qwen3 row gives the smallest mean-to-break-even ratio in the roster.

Model	Median request (ms)	Mean request (ms)	Break-even (ms)	Mean / break-even
Original panel				
Qwen3 4B	57.94	83.45	0.286	292×
Llama 3.2 3B	44.22	62.17	0.064	971×
Gemma 4 E2B	63.28	87.83	0.057	1,533×
Ministral 3 3B	47.93	70.27	0.102	689×
Phi-4 Mini 3.8B	52.01	72.73	0.055	1,315×
Granite 4.1 3B	54.14	74.12	0.063	1,179×
OLMo 3 7B	136.96	218.39	0.074	2,957×
Falcon-H1 7B	203.62	1,300.85	0.505	2,577×
New-family candidates				
SmolLM3 3B	48.18	64.12	0.066	966×
MiniCPM 4.1 8B	106.64	165.72	0.092	1,798×
GLM-4 9B	114.05	166.00	0.209	796×
Apertus 8B	101.84	158.01	0.091	1,744×

Even this closest case is 292 times above break-even, corresponding to QP overhead of about 0.34% of the estimated avoided Qwen3 inference time. Models with slower marginal inference would have a larger margin if their stopping trajectory were held fixed. Absolute timings and the ratio are machine-, batching-, request-order-, and implementation-dependent, whereas candidate calls remain the more portable primary measure. The current requests also generate only one constrained token; longer outputs or inference-time reasoning would change the inference side of the comparison, not the QP program itself.

D.6 Provenance and sensitivity checks

The supplementary manifests bind the configurations, task and model sources, repository revisions, prediction shards, output filenames, and SHA-256 hashes. The loss sampler and matched stopping comparison are post-hoc cache replays; they do not run new inference. The exact variance audit is also cache-only and adds no refit, stopping rule, or outcome bootstrap. The model-family cache and run namespace is separate from the original panel, and the four new candidates are evaluated one at a time.

Before the model-family run, a compatibility preflight scored WiC[0] for SmolLM3 and Apertus and issued diagnostic score requests for MiniCPM. GLM was tested only on synthetic input. The preflight computed no usefulness, QP, aggregate, or method result and changed none of the roster, tasks, methods, budgets, threshold, or success rules.

Thirteen of 60 model-panel held-out usefulness labels change on the smaller 63/64-row held-out sensitivity set. Excluding QASC leaves the LLM headline conclusions unchanged. Benchmark contamination in model pretraining cannot be ruled out, and the model-family test reuses the task surface from the task-family experiment. These checks limit the claim to the fixed artifacts, tasks, and reductions in queries to the screening set reported here.

E SAMPLING AND STOPPING CONTROLS

The supplementary accuracy results in this appendix measure agreement with the held-out usefulness label. Matched-cap savings isolate early stopping using Equation (52); the main figures use the full screening set as their reference.

E.1 QP AND COMBINED STOPPING

On the frozen paths, combining the QP and alignment-completion bounds gives the same calls and decisions as QP-only at every reported budget. At the largest cap, the QP rejects earlier or uniquely while saving at least one call on 4,511 regression paths, 512 unseen-task paths, and 237 unseen-model paths; alignment does so on none. This is an empirical ordering of the implemented bounds. Equation (37) orders exact bounds when they use the same completion set.

E.2 INCUMBENT-LOSS AND ALIGNMENT QUERY PATHS

The main sampling comparison disables stopping. A separate replay asks whether its conclusion changes after both schedules use the same QP-only certificate. Table 8 reports the largest common cap; positive call differences mean that the incumbent-loss path uses more calls.

Table 8: Post-hoc query-schedule comparison with the coordinate-box QP held fixed. Accuracy scores held-out usefulness labels. Values are paired point differences with 95% intervals. No accuracy contrast is resolved. The alignment schedule uses modestly fewer calls in regression; the LLM call differences are unresolved.

	Loss − alignment	
Panel and cap	held-out accuracy (pp)	Loss − alignment calls
Regression, 2048	$+0.01\ [-0.05, 0.09]$	$+5.61\ [2.44, 9.35]$
LLMs: unseen tasks, 192	$0.00\ [0.00, 0.00]$	$-1.20\ [-2.91, 0.24]$
LLMs: unseen models, 192	$0.00\ [0.00, 0.00]$	$-0.11\ [-2.03, 1.55]$

This check supports using the frozen alignment schedule for the matched stopping comparison, but not a claim that it is universally better. The two schedules have similar accuracy, and their relationship is already explained by Proposition 1.

E.3 RESIDUAL-AWARE ASSIGNMENT PLACEBO

A second sampling check asks whether residual-aware row assignment matters relative to a matched permutation. It deterministically reassigns complete alignment-bound pairs while preserving their multiset, the four stratum sizes, the stabilized allocation, nested budgets, and the post-stratified convex refit. No method in this ablation stops early.

The assignment effect has the same sign in all three panels, but only the regression interval excludes zero. Placebo minus uniform is -1.53 points in regression, $+0.50$ on unseen task families, and -0.50 on unseen model families; all but the regression contrast have intervals that include zero. Unlike the incumbent-loss comparison, this placebo does not distinguish alignment geometry from a natural incumbent-difficulty score. It reuses the frozen panels and is not an independent confirmation.

Table 9: Post-hoc row-assignment placebo at 64 calls. Accuracy scores held-out usefulness labels and is reported in percent; effects are percentage-point differences. The contrast isolates the row-to-bound assignment inside the fixed four-stratum, weighted-refit pipeline. It does not compare alignment with every possible row-difficulty score.

Panel	Uniform	Placebo	Alignment-stratified	Assignment effect [95% interval]
Regression	79.22	77.69	81.88	$+4.19\ [2.75, 5.63]$
LLMs: unseen tasks	76.83	77.33	81.50	$+4.17\ [-0.92, 9.33]$
LLMs: unseen models	78.17	77.67	81.00	$+3.33\ [-2.34, 8.83]$

E.4 Other screening controls

The LLM-only scalar mixture freezes the incumbent aggregate fitted on the full screening set and fits one mixing weight between it and the candidate on a uniform prefix. At 64 calls it reaches 75.67% held-out usefulness-label accuracy on unseen task families and 74.00% on unseen model families, compared with 76.83% and 78.17% for uniform full-simplex refitting. The model-gate difference is -4.17 points with interval $[-7.00, -1.50]$. This is a useful simple baseline, but it is too restrictive when the incumbent weights should also move.

A standalone optimistic QP queries rows in descending order of their known alignment-interval width and continues until certified rejection or the exact decision from the full screening set. On the regression panel it reproduces every decision from the full screening set at 1,356 mean calls. This gives one naturally stopping reference point rather than a budget curve.

E.5 Supplementary held-out diagnostics

The primary comparison of screening accuracy is complemented by the held-out usefulness label defined in Equation (49). In the original LLM panel, every model is useful on at least one task family, and 29 of the 36 useful additions have worse standalone Brier loss than their incumbent aggregate (Table 4).

With early stopping disabled, the two scores improve held-out decision accuracy at 64 calls by 2.66–3.16 percentage points in regression and 3.92–4.67 points on unseen LLM tasks; all four paired 95% intervals exclude zero. The unseen-model estimates are less precise: the loss-minus-uniform interval touches zero and the alignment-minus-uniform interval crosses it. Direct loss–alignment contrasts at 64 calls remain unresolved: $+0.50$ points $[-0.25, 1.22]$ in regression, -0.75 $[-3.08, 1.42]$ on unseen tasks, and $+1.83$ $[-1.17, 5.00]$ on unseen models.

At the largest caps, QP and capped no stopping have the same held-out usefulness-label accuracy: 89.24%, 85.00%, and 83.33% in regression, unseen LLM tasks, and unseen LLM models. Relative to QP, two-negative stability changes this accuracy by -8.10 points $[-11.22, -4.94]$, -2.75 $[-6.17, 0.83]$, and -3.83 $[-8.67, 0.83]$, respectively. The LLM intervals include zero. Posterior scenarios at $\delta = 0.05$ change held-out accuracy by -4.17, 0.00, and -0.17 points relative to capped no stopping (Table 10).

The decision from the full screening set itself disagrees with held-out usefulness in 150 of 1,400 regression cases. It misses 7 of 36 held-out-useful additions on unseen LLM tasks and 5 of 20 on unseen models, giving useful-candidate recall of 80.6% and 75.0%. It also accepts 11 of 84 held-out-negative task-panel cases and 5 of 40 held-out-negative model-panel cases. The certificate cannot repair this gap between full-evaluation decisions and held-out usefulness. Thirteen of 60 model-panel labels change on the smaller held-out sensitivity split; intervals for held-out accuracy condition on the observed labels and exclude this label uncertainty.

For completeness, largest-cap QP savings relative to capped no stopping are 5.89%, 23.03%, and 21.42% overall, and 11.52%, 34.54%, and 32.13% among paths for candidates rejected by full evaluation. Alignment completion saves 0.81%, 12.90%, and 11.17% overall against the same capped baseline. At the 192-row LLM census, per-candidate QP savings range from 4.44% (Falcon-H1)

to 32.89% (Phi-4 Mini) in the task panel, and from 10.22% (GLM-4) to 30.00% (SmolLM3) in the model panel. These are descriptive ranges without candidate-level uncertainty estimates.

F Posterior-scenario construction and diagnostics

F.1 Scenario decision rule

Let $\Pi_q(\cdot \mid \mathcal{O}_q)$ denote a working predictive distribution over complete candidate outputs after the queried observations $\mathcal{O}_q$. For a draw Z, let $\Delta^\star(Z)$ be its gain on the screening set after exact augmented convex reaggregation. Assuming $R_n(f^\star_{A,n}) > 0$, define $G^\star(Z) = \Delta^\star(Z)/R_n(f^\star_{A,n})$ and

$$p_q = \Pr_{Z\sim\Pi_q}\{G^\star(Z) \geq \gamma \mid \mathcal{O}_q\}, \qquad \gamma = 0.01. \tag{59}$$

At a rejecting checkpoint, all $N_{\mathrm{MC}} = 512$ scenarios are evaluated. If K_q attain the material gain, compute the one-sided 99% Clopper–Pearson upper bound (Clopper & Pearson, 1934) $U_{0.99}(K_q, N_{\mathrm{MC}})$ and reject when

$$U_{0.99}(K_q, N_{\mathrm{MC}}) < \delta. \tag{60}$$

A checkpoint can declare non-rejection early once the material count makes this inequality impossible. Failed refits count as material. The binomial bound controls finite simulation error at one checkpoint under the working distribution. It supplies neither coverage under model misspecification nor a time-uniform guarantee for repeated inspections.

F.2 Working predictive distributions

The scenario rule uses a working predictive model for missing outputs; it leaves the frozen alignment-stratified path, checkpoints, one-percent material-gain threshold, and exact ensemble refit unchanged. It is inactive through 64 calls and can therefore first change a reported decision at the next budget. For regression, the same random numbers are reused across checkpoints and values of δ on each path. The LLM replay uses checkpoint-specific draws and shares them only across values of δ within that checkpoint. Observed candidate outputs are copied unchanged into every completion.

For regression, write the candidate prediction as the incumbent aggregate plus a candidate deviation. This deviation follows a Bayesian linear model with 11 known row features: four frozen-stratum indicators, the standardized empirical residual, and six standardized incumbent deviations from their aggregate. The coefficient prior has mean zero and precision $4I$ conditional on the noise variance, while that variance has inverse-gamma parameters $(\alpha_0, \beta_0) = (2, 16/3)$. Each scenario shares one coefficient and variance draw across the screening set, draws the remaining row noise conditionally independently, inflates both posterior scales by 1.3, and clips the resulting predictions to $[-4, 4]$. Clipping approximates the original clipped prediction pipeline; it is not a censored-data posterior.

For the LLM panels, candidate probabilities are compared with the known incumbent aggregate in centered-log-ratio coordinates, using 10^{-4} additive smoothing. Within each frozen stratum, separate scalar normal–inverse-gamma models are updated for the target coordinate and the pooled non-target coordinates. Their prior parameters are $(\mu_0, \kappa_0, \alpha_0, \beta_0) = (0, 4, 2, 4)$, with the predictive scale again inflated by 1.3. A scenario shares its role–stratum location and variance draws, adds conditionally independent coordinate noise, and applies a softmax to return each row to the probability simplex. The model thus learns conditional location and scale, but not an arbitrary deviation shape or dependence structure. The separate scalar updates use a working independence approximation for candidate deviations in centered-log-ratio coordinates, whose coordinates sum to zero within a row. Softmax ensures valid probability vectors; the resulting predictive distribution is not claimed to be calibrated.

For $N_{\mathrm{MC}} = 512$ draws, the one-sided 99% Clopper–Pearson condition in Equation (60) allows at most 0, 6, 14, 35, and 81 material scenarios for $\delta = 0.01, 0.03, 0.05, 0.10$, and 0.20, respectively. Any failed refit is counted as material. The full replays had no solver failures: regression evaluated 24,942,080 complete scenarios with a batched exact active-face solver and checked 83,256 fits independently with SLSQP; the LLM replay evaluated 1,075,946 of 2,249,216 requested scenarios, with early abort accounting for the difference.

F.3 BUDGET TRAJECTORIES AND THRESHOLD SENSITIVITY

Figure 2 (b) compares screening accuracy and calls saved relative to the full screening set across budgets at the representative post-hoc threshold $\delta = 0.05$. The rule follows the safe QP through 64 calls by construction, so only the 128- and 192-call LLM budgets can differ. Once active, scenario call savings lie between QP and stability. The corresponding screening accuracy exceeds that of a pointwise randomized mixture of those two rules matched to the scenario mean calls: the descriptive difference ranges from 0.62 to 4.66 points in regression, 2.28 to 2.86 on unseen tasks, and 2.47 to 3.17 on unseen models. The mixing weight varies with budget, and these contrasts have no uncertainty intervals. They establish neither calibration nor population-level dominance.

Table 10 gives the complete threshold sweep at each panel's largest cap. Figure 4 then visualizes its matched-cap call savings and changes in screening accuracy. These supplementary plots isolate stopping relative to the same capped no-stop procedure, using Equations (52) and (53).

Combining Figure 4 with the held-out column of Table 10, all five LLM threshold points lie above the matched safe–stability mixture in both screening accuracy and held-out usefulness-label accuracy. Four of five regression points do so; $\delta = 0.20$ falls just below the mixture. This is an empirical frontier on frozen paths, not posterior-calibration evidence.

The literal marginal-box diagnostic instead puts a central posterior interval around each unseen coordinate and gives the optimistic QP the Cartesian product of those intervals. At $\delta = 0.05$, matched-cap savings are 6.32%, 23.08%, and 21.42% in the three panels, compared with 5.89%, 23.03%, and 21.42% for the safe box; neither policy changes agreement relative to capped no stopping. The intervals remain wide: their average width is 4.627 on the eight-unit regression support and 0.842 on the unit LLM support, and 45.7% of LLM role–stratum intervals remain the complete $[0, 1]$ box. Moreover, the optimistic QP may combine all individually favorable endpoints into a jointly unlikely completion. The comparison is not a one-factor ablation: the marginal and scenario methods also use different posterior parameterizations. Their numerical δ values therefore should not be interpreted as equivalent guarantees.

Table 10: Full conditional-scenario sweep at each panel's largest cap. Savings and agreement changes are relative to capped no stopping; Δ held-out scores usefulness-label accuracy. "New false rejects" counts paths where full evaluation and capped no stopping keep the candidate but scenarios reject it; these are additional final false rejections, not all strict early rejections. Denominators are 14,000, 1,200, and 600 paths. At $\delta = .05$, regression has 1,008 new final false rejections but 1,016 strict early false rejections. The values of δ were swept post hoc and carry model-based, pointwise semantics.

Panel	δ	Calls saved vs. capped no-stop (%)	Δ screening accuracy (pp)	Δ held-out (pp)	New false rejects
Regression	.01	41.70	−3.56	−1.91	503
	.03	48.36	−5.63	−3.21	794
	.05	51.82	−7.16	−4.17	1,008
	.10	57.13	−10.01	−5.90	1,408
	.20	64.16	−15.09	−8.67	2,119
LLM tasks	.01	32.61	−0.17	+0.00	2
	.03	35.72	−0.33	+0.17	4
	.05	37.21	−0.50	+0.00	6
	.10	39.89	−1.42	−0.08	17
	.20	42.17	−2.50	−0.83	30
LLM models	.01	31.92	−0.17	−0.17	1
	.03	35.17	−0.50	−0.50	3
	.05	36.56	−0.83	−0.17	5
	.10	38.97	−2.17	−0.50	13
	.20	41.78	−3.33	−0.33	20

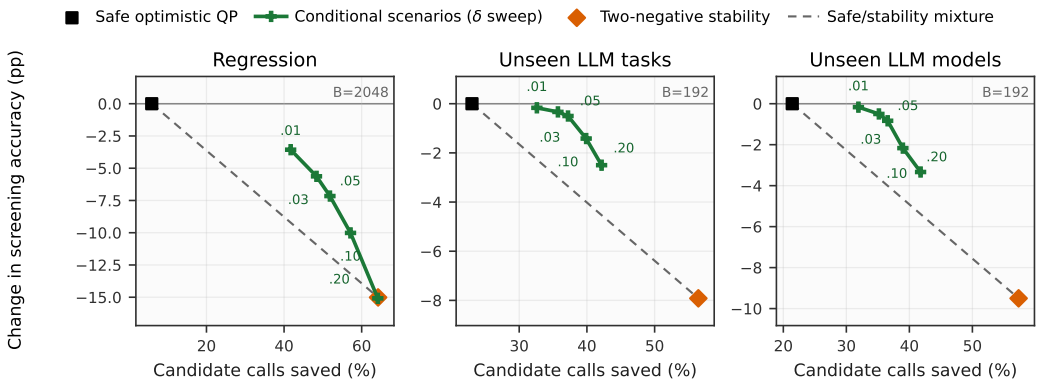

Figure 4: **Largest-cap posterior threshold sweep.** Each panel uses the same frozen alignment-stratified paths. Both axes compare against capped no stopping: call savings and screening accuracy changes. Black squares are the safe QP, green connected points are conditional posterior scenarios for $\delta \in \{0.01, 0.03, 0.05, 0.10, 0.20\}$, and orange diamonds are two-negative stability. Higher and farther right is better. The gray dashed segment joins the expected outcomes of randomized mixtures of the two endpoints. The scenario points are model-based and are not certified; path repeats are repeated measurements rather than independent cases.