BAYESIAN POST TRAINING ENHANCEMENT OF REGRESSION MODELS WITH CALIBRATED RANKINGS

Anonymous authors
Paper under double-blind review

ABSTRACT

Accurate regression models are essential for scientific discovery, yet high-quality numeric labels are scarce and expensive. In contrast, rankings (especially pairwise) are easier to obtain from domain experts or artificial intelligence (AI) judges. We introduce Bayesian Enhancement with Calibrated Ranking (BAYES-ECR), a novel plug-and-play method that improves a base regressor's prediction for a query by leveraging pairwise rankings between the query and reference items with known labels. BAYES-ECR performs a Bayesian update that combines a Gaussian likelihood from the regressor and the Bradley-Terry likelihood from the ranker. This yields a strictly log-concave posterior with a unique maximum likelihood solution and fast Newton updates. We show that prior state-of-the-art is a special case of our framework, and we identify a fundamental failure mode: Bradley-Terry likelihoods suffer from scale mismatch and curvature dominance when the number of reference items is large, which can degrade performance. From this analysis, we derive a calibration method to adjust the information originating from the expert rankings. BAYES-ECR shows a stunning 97.65% median improvement across 12 datasets over previous state-of-the-art method using a realistically-accurate ranker, and runs efficiently on a consumer-grade CPU.

1 INTRODUCTION

Regression underpins many scientific and engineering pipelines. In materials science and drug discovery, for example, structure-to-property predictors (i.e., through regression) can guide inverse molecular design loops by replacing expensive experiments and simulations with fast surrogates (Sanchez-Lengeling & Aspuru-Guzik, 2018; Liang et al., 2021; Allen & Tkatchenko, 2022). Yet, in many domains, collecting accurate absolute labels (e.g., molecule x is toxic at 1 nM) is slow and expensive (Wijaya et al., 2024), leading to data scarcity and limiting the accuracy of learning-based regressors.

An alternative to absolute labels is pairwise ranking (e.g., is molecule x more toxic than molecule y?). Pairwise rankings can be collected from human experts or from computational models, including general-purpose large language models (LLMs), at relatively low cost and with competitive accuracy. For humans, pairwise ranking reduces cognitive load (Routh et al., 2023) and mitigate scale-interpretation bias (Hoeijmakers et al., 2024) compared to directly predicting the absolute labels. Similarly, LLMs exhibit strong pairwise ranking ability in technical domains (Guo et al., 2023; Sun et al., 2025). This raises a central question: **can we enhance a pretrained regressor using only a handful of absolute labels plus inexpensive pairwise rankings without retraining the regressor?**

We propose Bayesian Enhancement with Calibrated Ranking (BAYES-ECR), a post-training enhancement method that combines a regressor's prediction with *calibrated* expert rankings via Bayesian inference. Given a pretrained regressor and an external pairwise ranker that compares a query item to a small set of labeled references, BAYES-ECR forms a posterior over the unknown scalar by combining (i) a Gaussian likelihood centered at the regressor's prediction, and (ii) a Bradley-Terry likelihood for pairwise rankings. We show that the posterior is strictly log-concave, yielding a unique solution and enabling fast optimization.

An additional key insight is that the default Bradley-Terry model can be mismatched to the behavior of pairwise rankers, leading to biased estimates and increased errors. BAYES-ECR addresses this

with (i) a learned temperature that calibrates the sigmoid slope of the Bradley-Terry model, and (ii) an accuracy-aware soft gate that regulates the influence of the ranker.

Specifically, our contributions are:

1. We introduce BAYES-ECR which combines a regressor likelihood with a calibrated ranker likelihood to enhance predictions without retraining. We prove that the state-of-the-art, RankRefine (Wijaya et al., 2025), is a special case under the Gaussian assumption.
2. We show how an uncalibrated Bradley-Terry model can bias the estimates when ranker likelihood dominates, and we provide a temperature calibration with accuracy-aware soft gating mechanism to mitigate the issue.
3. Across 12 cross-domain datasets (including real-world molecular datasets), BAYES-ECR significantly improves over existing post-training enhancement methods (Wijaya et al. (2025); Yan et al. (2024), and 3 other baselines) and remains effective when using imperfect LLM rankers. Specifically, BAYES-ECR achieves 19.33% median MAE reduction relying on 30 reference samples and a ranker with 65% accuracy, which translates to a stunning 97.65% relative improvement compared to RankRefine's 9.78% median MAE reduction.

Together, these results position BAYES-ECR as a practical and principled way to leverage readily-available pairwise information to enhance scalar regressor in data-scarce domains. Source code will be made public upon publication.

2 BACKGROUND

Pairwise Comparison Models. In pairwise (binary) comparisons, the probability of item x_i is preferred to x_j is often modeled as $P(x_i \succ x_j) = F(y_i - y_j)$, where y are latent scores and F is a cumulative distribution function (Cattelan, 2012). Classic pairwise comparison models include Thurstone-Mosteller with a Gaussian *link* function (Thurstone, 1927; Mosteller, 1951) and Bradley-Terry with a logistic *link* function (Bradley & Terry, 1952). These models provide a one-dimensional likelihood for the unknown scalar label that can be estimated using MAP or MLE estimation.

LLM-as-a-judge. General-purpose LLMs have demonstrated strong performance on relative comparisons across various domains (Qin et al., 2024; Wu et al., 2024; Guo et al., 2023), with evidence of better performance in pairwise comparison compared to score prediction (Zheng et al., 2023). With widely available web APIs (OpenAI, 2025; Anthropic, 2025; Google, 2025), collecting comparisons at scale is increasingly easy, making LLMs a convenient external ranker when numeric absolute labels are scarce.

Regression Refinement using Pairwise Rankings. Two representative approaches in this area are the Projection method (Yan et al., 2024) and RankRefine (Wijaya et al., 2025). Projection constrains the regressor's prediction to a feasible interval implied by non-contradictory pairwise outcomes. Meanwhile, RankRefine fuses the regressor's output with a rank-only estimate via inverse variance weighting. Our proposed method introduces a different paradigm: it reframes fusion as a Bayesian inference, where the regressor and ranker contribute likelihoods that jointly determine the posterior.

Relation to Learning from Human Feedback. Recent works on fine-tuning with human feedback (Christiano et al., 2017; Ziegler et al., 2019; Ouyang et al., 2022) also use pairwise rankings modeled via Bradley-Terry, but with a different goal. They (i) learn a global reward model and (ii) fine-tune a pretrained model accordingly. In contrast, we apply a per-query, post training prediction correction by combining the regressor's likelihood with a calibrated ranker likelihood. This produces a one-dimensional posterior without modifying the regressor's parameters.

3 METHOD

In this section, we describe the base formulation of BAYES-ECR (Section 3.1), along with the analyses of its behavior (Section 3.2) and proposed modifications for improving the performance (Section 3.3). We provide the detailed proofs of the analysis in the Appendix.

3.1 BAYESIAN INFERENCE ON EXPERT RANKINGS TO IMPROVE REGRESSION MODELS

Let f be a regressor trained on $\mathbb{C} = \{(x_j, y_j)\}_{j=1}^{N}$, with scalar labels $y_j \in \mathbb{R}$ and prediction $f(x_j) = \hat{y}_j^{\text{re}}$. Let R be an expert pairwise ranker that returns a binary comparison: $R(x_a, x_b) = 1$ if it predicts $y_a > y_b$, and $R(x_a, x_b) = 0$ otherwise. We are given a reference set $\mathbb{D} = \{(x_i, y_i)\}_{i=1}^{k}$ with known y_i, which can be, but not always is, the regressor's training set $\mathbb{C}$. For a query x_0 with unknown $y_0 \in \mathbb{R}$, we collect *expert rankings* against all references: $\mathbb{G} = \{r_i = R(x_0, x_i)\}_{i=1}^{k}$ where $r_i \in \{0, 1\}$.

Assuming conditional independence of regressor and ranker given the true y_0, the Bayes' rule yields

$$\underbrace{p(y_0 \mid \hat{y}_0^{\text{re}}, \mathbb{G})}_{\text{posterior}} \propto \underbrace{p(\hat{y}_0^{\text{re}} \mid y_0)\}}_{\text{reg. likelihood}} \underbrace{p(\mathbb{G} \mid y_0)}_{\text{rank likelihood}} \underbrace{p(y_0)}_{\text{prior}} \tag{1}$$

with Gaussian regressor likelihood $p(\hat{y}_0^{\text{re}} \mid y_0) = \mathcal{N}(\hat{y}_0^{\text{re}}; y_0, \sigma_{\text{re}}^2)$ and rank likelihood

$$p(r_i = 1 \mid y_0, y_i) = s\big(y_0 - y_i\big), \qquad s(z) = 1/(1 + e^{-z}), \tag{2}$$

$$p(\mathbb{G} \mid y_0) = \prod_{i=1}^{k} s\big(y_0 - y_i\big)^{r_i} \Big(1 - s\big(y_0 - y_i\big)\Big)^{1 - r_i}, \tag{3}$$

which is the product of all pairwise comparisons in $\mathbb{G}$ under the Bradley-Terry model (Bradley & Terry, 1952).

With the two likelihoods, the *base* formulation of BAYES-ECR enhances a regressor prediction with expert rankings using a maximum *a posteriori* (MAP) estimation (Murphy, 2022) which maximizes y with the objective function

$$\boxed{\mathcal{L}(y) = -\frac{1}{2\sigma_{\text{re}}^2}(\hat{y}_0^{\text{re}} - y)^2 + \sum_{i=1}^{k} \Big[r_i \log s(y - y_i) + (1 - r_i) \log(1 - s(y - y_i)) \Big] + \log p(y).} \tag{4}$$

If we assume a flat (uninformative) prior, i.e., $p(y) = 1$, BAYES-ECR reduces to a maximum likelihood estimation (MLE) (Murphy, 2022). The objective function of the BAYES-ECR (Equation 4) is strictly log-concave so we can obtain the maximum efficiently.

Lemma 3.1. (Strict log-concavity.) If $\sigma_{\text{re}}^2 > 0$ and $\log p(y_0)$ is concave (including the flat prior), then $\log p(y_0 \mid \hat{y}_0^{\text{re}}, \mathbb{G})$ is strictly concave in y_0. □

Corollary 3.2. (Existence and uniqueness.) The maximum *a posteriori* (and maximum likelihood under a flat prior) estimate exists and is unique. □

Proposition 3.3. (RankRefine (Wijaya et al., 2025) is a special case of BAYES-ECR under Gaussian assumption.) Let $\mathcal{L}_{BT}(y)$ denote the rank-only log-likelihood and $\hat{y}_0^{\text{ra}} = \arg\max_y \mathcal{L}_{BT}(y)$. By assuming Gaussianity on the ranker likelihood (similar to Wijaya et al. (2025)), the second-order expansion of $\mathcal{L}_{BT}(y)$ around $\hat{y}_0^{\text{ra}}$ yields

$$p(\mathbb{G} \mid y) \approx \mathcal{N}(y; \hat{y}_0^{\text{ra}}, \sigma_{\text{ra}}^2), \qquad \sigma_{\text{ra}}^2 = \Big[\sum_{i=1}^{k} s(\hat{y}_0^{\text{ra}} - y_i)(1 - s(\hat{y}_0^{\text{ra}} - y_i)) \Big]^{-1}. \tag{5}$$

Combining this with the Gaussian regressor likelihood and a flat prior gives the inverse-variance weighted (IVW) (Cochran & Carroll, 1953) estimator, i.e., RankRefine (Wijaya et al., 2025), for which the estimate $\hat{y}_0^{rr}$ is,

$$\hat{y}_0^{\text{rr}} = \frac{\hat{y}_0^{\text{re}}/\sigma_{\text{re}}^2 + \hat{y}_0^{\text{ra}}/\sigma_{\text{ra}}^2}{1/\sigma_{\text{re}}^2 + 1/\sigma_{\text{ra}}^2}. \quad \square \tag{6}$$

Proof. The gradient and curvature of $\mathcal{L}_{BT}(y)$ are

$$\mathcal{L}'_{BT}(y) = \sum_{i=1}^{k} \big(r_i - s(y - y_i)\big), \qquad \mathcal{L}''_{BT}(y) = -\sum_{i=1}^{k} \Big(s(y - y_i)(1 - s(y - y_i))\Big). \tag{7}$$

The second-order Taylor expansion of $\mathcal{L}_{BT}$ around $y = \hat{y}_0^{\text{ra}}$ is

$$\mathcal{L}_{BT}(y) \approx \mathcal{L}_{BT}(\hat{y}_0^{\text{ra}}) + \mathcal{L}'_{BT}(\hat{y}_0^{\text{ra}}) \cdot (y - \hat{y}_0^{\text{ra}}) + \tfrac{1}{2}\mathcal{L}''_{BT}(\hat{y}_0^{\text{ra}}) \cdot (y - \hat{y}_0^{\text{ra}})^2. \tag{8}$$

The first term is constant and the second term is zero, therefore,

$$\mathcal{L}_{BT}(y) \propto -\frac{1}{2}\sum_{i=1}^{k} s(\hat{y}_0^{\text{ra}} - y_i)(1 - s(\hat{y}_0^{\text{ra}} - y_i))(y - \hat{y}_0^{\text{ra}})^2. \tag{9}$$

Similar to RankRefine, we make a strong assumption of Gaussianity for the BT likelihood,

$$p(\mathbb{G} \mid y_0) \propto \exp(\mathcal{L}_{BT}(y)) = \exp\Big(-\frac{1}{2}\sum_{i=1}^{k} s(\hat{y}_0^{\text{ra}} - y_i)(1 - s(\hat{y}_0^{\text{ra}} - y_i))(y - \hat{y}_0^{\text{ra}})^2\Big) \equiv \mathcal{N}(\mu_{ra}, \sigma_{\text{ra}}^2), \tag{10}$$

$$\mu_{ra} = \hat{y}_0^{ra}, \qquad \sigma_{\text{ra}}^2 = \Big[\sum_{i=1}^{k} s(\hat{y}_0^{\text{ra}} - y_i)(1 - s(\hat{y}_0^{\text{ra}} - y_i))\Big]^{-1}.$$

The joint likelihood of the conditionally independent regressor and ranker is now a product of two Gaussians. The MLE, i.e., MAP estimate with a flat prior, then solves a weighted least-squares problem whose maximizer is the inverse-variance weighted average. □

3.2 Analysis of Potential Performance Degradation of Base Bayes-ECR

Under the base formulation, Bayes-ECR can degrade when the reference size k and ranker accuracy a exceed certain thresholds. The effect is dataset-dependent and arises from a mismatch between the Bradley-Terry model and the behavior of real-world rankers.

Separating the rank likelihood from the objective in Equation 4 and maximizing it yields the rank-only MLE $\hat{y}_0^{\text{ra}}$. If the ranker were perfectly accurate and the likelihood perfectly specified, then $\hat{y}_0^{\text{ra}} \in (y_m, y_{m+1})$, where $y_m < y_0 < y_{m+1}$ are the closest references and $m = \sum_i r_i$ is the number of references ranked below y_0. In practice, the sigmoid in the Bradley-Terry model induces a *soft-. vs. hard-count* mismatch: the rank-only target solves a soft-count equation and need not lie in (y_m, y_{m+1}).

Lemma 3.4 (Rank-only MLE may target a pseudo ground truth.) Let $u_i(y) = s(y - y_i)$ and $m = \sum_{i=1}^{k} r_i$. The Bradley-Terry log-likelihood $\mathcal{L}_{BT}(y)$ is strictly concave and its unique maximizer $\hat{y}_0^{\text{ra}}$ satisfies

$$\mathcal{L}'_{BT}(\hat{y}_0^{\text{ra}}) = 0 \iff \sum_{i=1}^{k} u_i(\hat{y}_0^{\text{ra}}) = m. \tag{11}$$

Because m is a *hard* count while u_i are sigmoid *soft* counts, the solution to $\sum_{i=1}^{k} u_i(\hat{y}_0^{\text{ra}}) = m$ is a pseudo target $\tilde{y}_0$, which can lie outside (y_m, y_{m+1}) even under a perfectly accurate ranker, biasing $\hat{y}_0^{\text{ra}}$. See Figure 1 for illustration. □

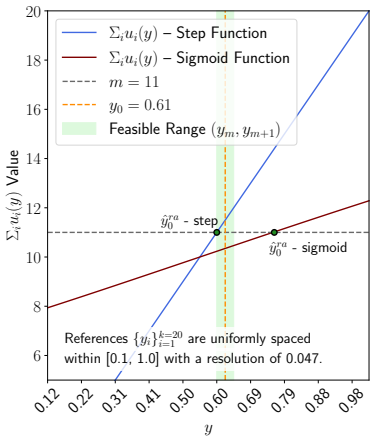

Figure 1: Soft- vs. hard-count mismatch from Lemma 3.4. We generate a synthetic reference set $\{y_i\}$ and ground truth y_0. From Equation 11, the rank-only MLE $\hat{y}_0^{ra}$ is the intersection between $\sum u_i(y)$ and m, and (y_m, y_{m+1}) forms the feasible range. We can see that $\sum u_i(y)$ with a step function correctly intersects the feasible range. However, using a sigmoid function results in a biased intersection outside the feasible range.

Bias from the rank-only target can increase the overall error if the rank likelihood dominates the regressor likelihood. Rank likelihood dominance is governed by its curvature (Fisher information (Ly et al., 2017)), which grows with k.

Lemma 3.5. (Rank curvature (Fisher information) scales with k.) The Fisher information of the rank log-likelihood is $I_{\text{rank}}(y) = \sum_{i=1}^{k} u_i(y)(1 - u_i(y))$, so $I_{\text{rank}}(y)$ grows linearly with k. Moreover, $0 \le I_{\text{rank}}(y) \le \frac{k}{4}$. □

Lemma 3.6. (Info-weighted Newton step and dominance.) As k grows, the optimization process using Newton's method (Murphy, 2022) to solve Equation 4 is dominated by the rank likelihood term. That is, denoting $g_{\text{tot}}(y)$ and $I_{\text{tot}}(y)$ as the total gradient and Fisher information of both likelihoods at y, I_{reg} as the Fisher information of the regressor likelihood, and $\tilde{y}^{ra}(y)$ as the target of the ranker likelihood term at y, a Newton step is

$$y \leftarrow y + \frac{g_{\text{tot}}(y)}{I_{\text{tot}}(y)} = \frac{I_{\text{reg}} \cdot \hat{y}_0^{\text{re}} + I_{\text{rank}}(y) \cdot \tilde{y}^{\text{ra}}(y)}{I_{\text{reg}} + I_{\text{rank}}(y)}, \tag{12}$$

which is an information-weighted average using Fisher information as the weights. □

Since $I_{\text{reg}} = 1/\sigma_{\text{re}}^2$ is constant, Lemma 3.5 implies the rank likelihood term eventually dominates as k grows. Apart from Newton, for a first-order step $y \leftarrow y + \eta g_{\text{tot}}(y)$, the same decomposition yields $y \leftarrow y + \eta[I_{\text{reg}}(\hat{y}_0^{\text{re}} - y) + I_{\text{rank}}(y)(\tilde{y}^{\text{ra}}(y) - y)]$, showing similar information-weighted pull. The dominance of the rank likelihood then follows from Lemma 3.5.

Dominance alone is harmless, but combined with Lemma 3.4, a biased rank likelihood target $\tilde{y}_0$ can steer the refinement away from y_0. Note that dominance occurs only when both k and the ranker accuracy a are sufficiently large. Moreover, the accuracy threshold decreases with k.

Lemma 3.7. (Accuracy-k threshold for rank dominance.) Define

$$p^* = \frac{1}{k}\sum_{i=1}^{k} \mathbb{1}(y_0 > y_i), \qquad \hat{p}(y) = \frac{1}{k}\sum_{i=1}^{k} u_i(y). \tag{13}$$

Let $a \in [0.5, 1]$ denote the ranker accuracy. Then, at any iterate y, the rank likelihood dominates the regressor likelihood whenever

$$a > \frac{1}{2} + \frac{|y - \hat{y}_0^{\text{re}}|}{2k\sigma_{\text{re}}^2 |p^* - \hat{p}(y)|}. \quad \square \tag{14}$$

The right hand side of the inequality is the threshold accuracy $a_{\text{thr}}(y)$, and it decreases as $1/k$ for a fixed y, σ_{re}^2, and $|p^* - \hat{p}(y)| > 0$. Therefore, the required ranker accuracy such that the rank term dominates the Newton step shrinks as k grows.

Corollary 3.8. (Rank likelihood can degrade base BAYES-ECR.) Under the base formulation of BAYES-ECR, the rank-only target can be biased (Lemma 3.4). This may lead to a performance degradation when the ranker likelihood term dominates (Lemma 3.6). In general, ranker likelihood dominance grows with the reference set size k and the ranker accuracy a. When k is large, the required a for the ranker likelihood to dominate is lower (Lemma 3.7).

3.3 BAYES-ECR WITH CALIBRATED EXPERT RANKINGS

The degradation stems from the Bradley-Terry modeling mismatch (Lemma 3.4): when many pairwise gaps $(y_a - y_b)$ fall on the sigmoid's transition region rather than its saturated tails, the rank-only target becomes biased. This is a *unit-scale* issue; datasets with small average $|y_a - y_b|$ have higher probability of placing many pairs on the transition slope.

We address this by introducing a temperature τ to adjust the slope, i.e., replace $u_i(y) = s(y - y_i)$ with $v_i(y;\tau) = s((y - y_i)/\tau)$. This (i) aligns the logistic slope with label units so that the tempered score $\sum_i v_i(\hat{y}_0^{\text{ra}};\tau)$ can match the observed count $m = \sum_i r_i$, removing the *soft- vs. hard-count* mismatch, and (ii) sets the rank curvature to

$$I_{\text{rank}}(y;\tau) = \tau^{-2} \sum_i v_i(y;\tau)\big(1 - v_i(y;\tau)\big), \tag{15}$$

thereby controlling rank dominance at large k. We estimate τ for a dataset via a one-parameter logistic fit on the reference set, thus requiring no extra labeled data,

$$\Pr(r = 1 \mid y_a, y_b; \hat{\omega}) = s(\hat{\omega}(y_a - y_b)), \qquad (y_a, y_b) \in \mathbb{D}, \tag{16}$$

and set $\tau = \hat{\tau}_{\text{cal}} = 1/\hat{\omega}$. With this calibration of expert rankings, the Newton update is an information-weighted average with temperature-controlled curvature,

$$\boxed{y \leftarrow \frac{I_{\text{reg}} \cdot \hat{y}_0^{\text{re}} + I_{\text{rank}}(y;\tau) \cdot \tilde{y}^{\text{ra}}(y;\tau)}{I_{\text{reg}} + I_{\text{rank}}(y;\tau)}}. \tag{17}$$

where $\tilde{y}^{\text{ra}}(y;\tau) = y + \frac{\sum_{i=1}^{k}(r_i - v_i(y;\tau))}{I_{\text{rank}}(y;\tau)}$.

Setting $\tau < 1$ increases the rank likelihood curvature (via the τ^{-2} factor) and shifts the update toward the rank target. When the ranker is accurate, this permits rank dominance *without* the bias highlighted by Lemma 3.4, improving performance over the naive BAYES-ECR.

On the other hand, when the ranker is less accurate, overly large curvature would let noisy rank signals dominate and harm performance. We therefore apply an *accuracy-aware soft gate*:

$$\tau(a) = 1 + (\hat{\tau}_{\text{cal}} - 1)(w(a))^{\gamma}, \qquad \gamma \geq 1, \tag{18}$$

$$w(a) = \max(0, 2a - 1) \in [0, 1]. \tag{19}$$

Thus $\tau(a) \approx 1$ when $a \approx 0.5$, $\tau(a) \to \hat{\tau}_{\text{cal}}$ as $a \to 1$, and intermediate a produces a smooth interpolation. The final BAYES-ECR with calibrated expert rankings is summarized in Algorithm 1.

Algorithm 1 BAYES-ECR Algorithm

Inputs: x_0, $\hat{y}_0^{\text{re}}$, σ_{re}^2, $\mathbb{G} = \{(x_i, y_i)\}_{i=1}^{k}$; $\mathbb{D} = \{y_i\}_{i=1}^{k}$ R; (optional: $p(y)$)
Collect Comparisons: $r_i \leftarrow R(x_0, x_i)$
Estimate Temperature: fit $s(\alpha(y_a - y_b))$ on labeled pairs from $\mathbb{D}$; set $\hat{\tau}_{\text{cal}} = 1/\hat{\alpha}$. Estimate the ranker accuracy a; set $\tau \leftarrow 1 + (\hat{\tau}_{\text{cal}} - 1)(\max(0, 2a - 1))^{\gamma}$.
MAP / MLE Optimization: Maximize Equation 4 with $s(\cdot)$ replaced by $s((\cdot)/\tau)$. With Newton steps described on Equation 17, iterate to convergence.
Output: $\hat{y}_0^{\text{rr}}$ and approximate uncertainty $\sigma_{\text{post}}^2 \approx (I_{\text{reg}} + I_{\text{rank}}(\hat{y}_0^{\text{rr}}; \tau))^{-1}$.

4 EXPERIMENTS

To demonstrate the benefits of BAYES-ECR in data-scarce domains, we evaluate on 9 molecular property prediction datasets from the TDC ADMET regression task (Huang et al., 2021): Caco-2 (Wang et al., 2016), Clearance Microsome and Clearance Hepatocyte (Di et al., 2012), log Half-Life (Obach et al., 2008), FreeSolv (Mobley & Guthrie, 2014), Lipophilicity (Wu et al., 2018), PPBR, Solubility (Sorkun et al., 2019), and VDss (Lombardo & Jing, 2016). For each experiment, we sample $N = 100$ molecules from the original training split and $L = 100$ molecules from the original test split. We repeat this train/test resampling for 10 random seeds, similar to a Monte Carlo cross-validation. The reference set $\mathbb{D}$ is sampled from the training set with $k \in \{3, 10, 20, 30, 50, 100\}$. Two types of base regressors are used: random forest (RF) (Ho, 1995) and multilayer perceptron (MLP) (Rumelhart et al., 1986), both trained on the N training labels. We also use three tabular regression datasets: crop-yield prediction from sensor data (Soundankar, 2025), student-performance prediction (Cortez, 2014), international-education cost estimation (Shamim, 2025).

We evaluate four BAYES-ECR variants: MAP with a Gaussian prior, MLE, MLE with temperature scaling (MLE-Temp), and MLE with temperature scaling and soft gating (MLE-GatedTemp). The full BAYES-ECR, i.e., MLE-GatedTemp, is the default variant we use in the experiments. We compare our method to two post-training regression enhancement baselines: (i) Projection-based approach (Yan et al., 2024) and (ii) RankRefine (Wijaya et al., 2025). Additionally, we adapt two pairwise ranking models to create rank-only regression enhancement baselines: (i) Bradley-Terry (rank-only MLE using the Bradley-Terry model), (ii) Thurstone (rank-only MLE using the Thurstone-Mosteller model (Thurstone, 1927; Mosteller, 1951)). Following Wijaya et al. (2025), we report $\beta \equiv \text{MAE}_{\text{post}}/\text{MAE}_{\text{base}}$, which is the ratio between the post-enhancement Mean Absolute Error (MAE) and the MAE of the regressor-only predictions (lower is better).

We use two types of rankers: (i) *oracle* ranker to study the effect of ranker accuracy, and (ii) LLM ranker to demonstrate real-world use cases. For the oracle ranker, we set $r_i = \mathbb{1}(y_0, y_i)$ and then flip the outcomes with probability $1 - a$ to simulate a ranker with accuracy $a \in [0.5, 1]$. For LLM rankers, we prompt publicly-available models to compare a list of pairs of molecular text representations (SMILES (Weininger, 1988)) and measure the accuracy a on a small validation set.

Additional discussions and experiments on LLM usage, temperature calibration, multi-target regression, and reference set size are available in Appendix.

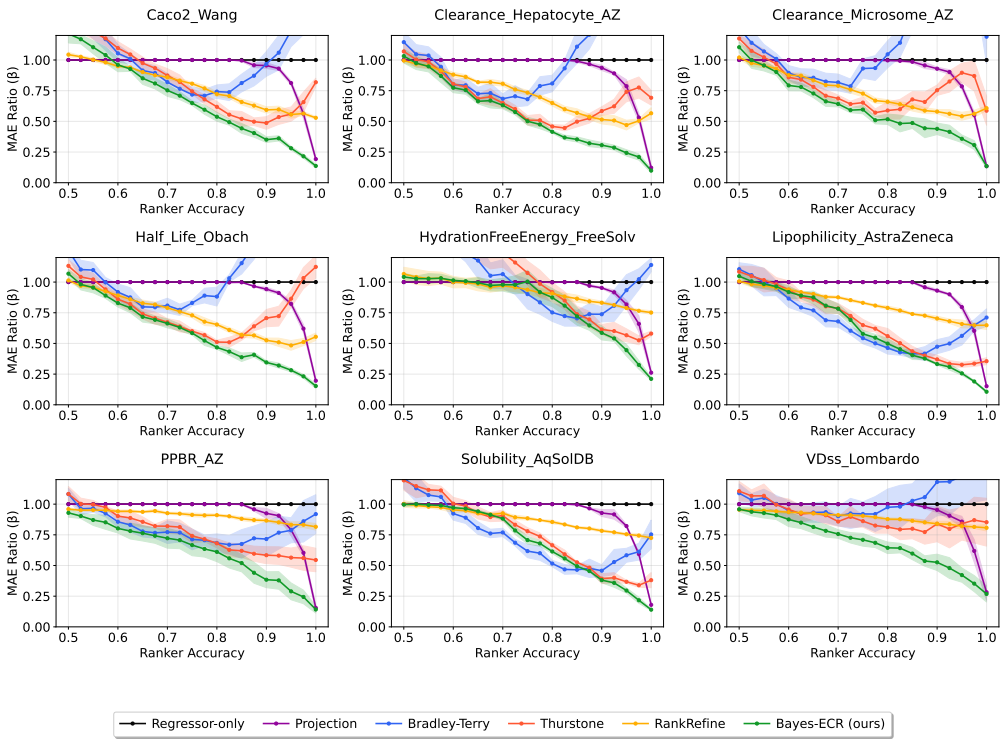

Figure 2: MAE ratio β (lower is better) as a function of oracle ranker accuracy a on nine TDC ADMET datasets ($k = 30$). We report the mean and standard deviation across 10 random splits. BAYES-ECR (MLE-GatedTemp variant) outperforms projection and RankRefine across ranker accuracies, with especially strong gains at moderate, real-world accuracy. Results for other k values with similar trends are available in Appendix.

4.1 MAIN RESULTS WITH ORACLE RANKERS

Across all nine ADMET datasets at $k = 30$, BAYES-ECR (MLE-GatedTemp variant) consistently outperforms previous state-of-the-art, RankRefine, and 3 other baselines (Figure 2). Gains over recent custom-built enhancement methods, RankRefine and Projection, are largest in the mid-accuracy regime (65% - 90%). At lower ranker accuracies (<65%), BAYES-ECR matches or exceeds RankRefine and clearly outperforms Projection. At high ranker accuracies, RankRefine performance plateaus, while BAYES-ECR continues to improve, converging to Projection's performance at perfect (yet unrealistic) ranker accuracy.

4.2 ABLATION: PRIOR, TEMPERATURE, AND ACCURACY-AWARE SOFT GATING

Figure 3 shows that MAP/MLE variants of BAYES-ECR can degrade as k grows, confirming that rank-dominance in the Newton step and the soft-hard count mismatch on the rank term can bias the enhancement (Corollary 3.8). The rank-only Bradley-Terry and Thurstone models in Figure 2 also exhibit similar degradation, validating that the source of degradation is the ranker likelihood. Interestingly, MAP degrades less than MLE, indicating that prior can act as a regularizer. As proposed in Lemma 3.7, the ranker accuracy threshold at which degradation appears shifts lower as k increases.

Our full method (MLE-GatedTemp variant) solves the performance degradation issue. Temperature scaling aligns the sigmoid slope to the label scale and controls the rank curvature, mitigating degradation at high ranker accuracy, but can worsen performance at low ranker accuracy (Figure 3, MLE-Temp). Adding accuracy-aware soft-gating removes this trade-off (Figure 3, MLE-GatedTemp), enabling full BAYES-ECR to deliver strong performance across the complete ranker accuracy range. Note that Figure 3 also confirms that our enhancement method works with a reference set as small as $k = 3$, which is highly practical in the real-world setting.

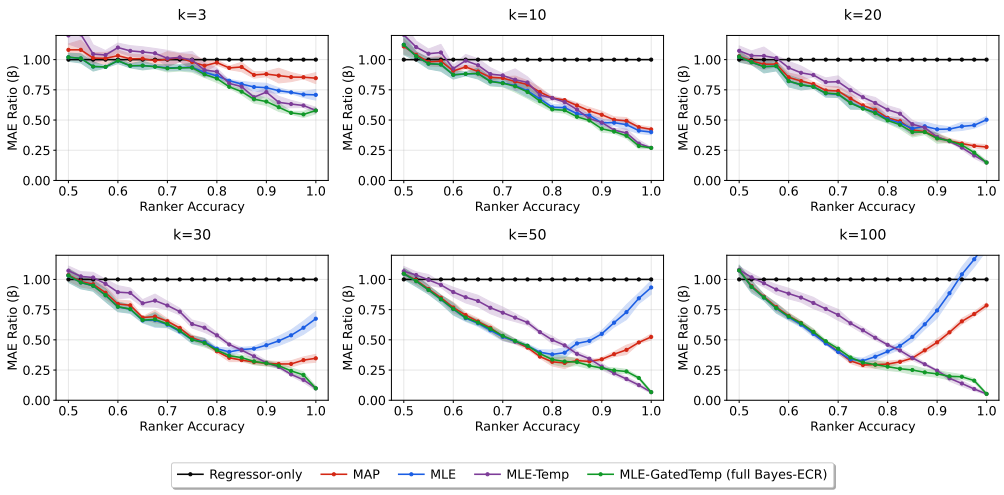

Figure 3: Effects of reference set size k and temperature scaling / soft gating on the Clearance Hepatocyte dataset. The shown methods are the four variants of BAYES-ECR, with *ours* being MLE-GatedTemp. We show the MAE ratio β (lower is better) as a function of oracle ranker accuracy a for $k \in \{3, 10, 20, 30, 50, 100\}$. Without temperature scaling, MAP and MLE variants degrade at larger k due to rank curvature dominance. Temperature (MLE-Temp) fixes the scale mismatch, and soft-gating (MLE-GatedTemp) further improves robustness when ranker accuracy is low.

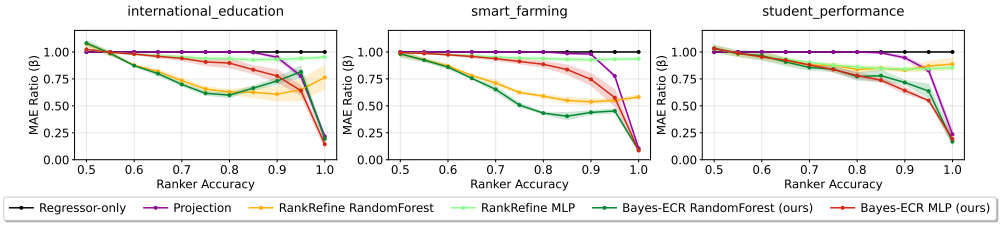

Figure 4: MAE ratio β on three non-molecular tabular datasets with Random Forest (RF) and Multilayer Perceptron (MLP) as base regressor models. We show that the refinements are generalizable to other domains and regressor models. Baseline (non-refined) MAEs are: **International Education Cost:** 0.0926 (MLP), 0.0609 (RF); **Smart Farming Yield:** 0.1448 (MLP), 0.1361 (RF); **Student Exam Performance:** 0.0727 (MLP), 0.0819 (RF).

4.3 CROSS-DOMAIN GENERALITY AND REGRESSOR DIVERSITY

To test cross-domain and cross-model generalization, we add three non-molecular tabular datasets and run both Random Forest (RF) and Multilayer Perceptron (MLP) regression models. Figure 4 shows that BAYES-ECR yields $\beta < 1$ across the ranker accuracy range, improving both RF and MLP models, and proving its generalization capability to other domains and regression models.

4.4 LLM AS IMPERFECT YET PRACTICAL RANKER

We replace the oracle ranker with off-the-shelf LLMs to obtain noisy but scalable expert rankings. With ChatGPT5 Thinking and Claude Sonnet 4 as rankers at $N = 50$ and $k = 20$, BAYES-ECR generally achieves the best β across five ADMET datasets. Detailed results are shown in Table 1.

4.5 RUNTIME ANALYSIS ON VARYING REFERENCE SET SIZE

Excluding the ranking predictions, BAYES-ECR runs in less than 1 ms on an Intel i7-13700 CPU when the reference set size $k \leq 1000$, covering realistic scenarios in data-scarce domains. The average running time to predict expert rankings using ChatGPT5-Thinking for 1000 queries and 50

Dataset Name	Half Life	FreeSolv	PPBR	Solubility	VDss
ChatGPT5 PRA (%)	62.20 ± 1.89	62.27 ± 2.00	63.29 ± 2.65	62.87 ± 3.27	65.95 ± 1.96
BAYES-ECR (β)	**0.952 ± 0.057**	**0.997 ± 0.021**	**0.928 ± 0.045**	**0.985 ± 0.015**	**0.853 ± 0.194**
RankRefine (β)	0.977 ± 0.024	1.002 ± 0.022	0.968 ± 0.016	0.985 ± 0.020	0.952 ± 0.055
Projection (β)	1.056 ± 0.171	1.119 ± 0.100	1.007 ± 0.026	0.992 ± 0.039	0.949 ± 0.082
Dataset Name	Half Life	FreeSolv	PPBR	Solubility	VDss
Claude4 PRA (%)	52.36 ± 2.22	72.47 ± 1.40	51.73 ± 3.25	60.41 ± 2.45	60.85 ± 2.12
BAYES-ECR (β)	**0.955 ± 0.058**	**0.977 ± 0.011**	**0.965 ± 0.042**	0.990 ± 0.010	**0.850 ± 0.191**
RankRefine (β)	0.981 ± 0.010	1.070 ± 0.052	0.979 ± 0.019	**0.988 ± 0.016**	0.951 ± 0.056
Projection (β)	1.124 ± 0.279	1.153 ± 0.126	1.031 ± 0.039	0.995 ± 0.014	0.959 ± 0.038

Table 1: Results using LLMs (ChatGPT5, Claude4) as external rankers, measured in β over ten train/test splits with $N = 50, k = 20$. PRA stands for pairwise ranking accuracy. Best result for each dataset-ranker is bolded. BAYES-ECR generally outperforms other refinement methods.

Dataset Name	Half Life	FreeSolv	PPBR	Solubility	VDss
$\Delta_y \cdot 10^{-7}$	0.028 ± 0.028	0.030 ± 5.71	8.51 ± 1.42	0.029 ± 1.33	3.24 ± 2.96

Table 2: Mean absolute difference between the refined predictions of RankRefine and BAYES-ECR with Gaussianity assumptions. The values are around the range of machine epsilon for a floating point precision, confirming that RankRefine is a special case of BAYES-ECR.

references (50,000 pairs in total) in parallel is 152.4 ± 54.7 seconds. Therefore, BAYES-ECR can be run efficiently on consumer-grade computes, and LLM inference is not a significant bottleneck.

4.6 RANKREFINE IS A SPECIAL CASE OF BAYES-ECR.

We showed that RankRefine (Wijaya et al., 2025) is a special case of BAYES-ECR in Proposition 3.3. We verify this empirically by comparing enhanced predictions from RankRefine and BAYES-ECR under the Gaussianity assumptions. Across five ADMET tasks, the mean absolute difference Δ_y is at the level of machine epsilon (Table 2), confirming that Rankrefine is a special case of BAYES-ECR with Gaussianity assumptions.

5 LIMITATIONS

- **Noise model for the oracle ranker.** We inject pairwise flips uniformly to reach the desired oracle ranker accuracy. Real rankers (human or LLM) might exhibit systematic biases, which could negatively affect the Bayesian inference at lower ranker accuracies.
- **Dependence on regressor uncertainty.** BAYES-ECR requires a base regressor that can quantify uncertainty, which many off-the-shelf regressors lack. A straightforward solution is to use ensembling or Monte Carlo dropout.
- **Reference-set selection.** Regression improvements may depend on the composition of the reference set (coverage, label diversity, resolution). Future works can explore more sophisticated, sequential sampling strategy to select the references.

6 CONCLUSION

We introduced BAYES-ECR, framing post training regression enhancement as a problem of Bayesian inference rather than heuristic fusion. This perspective not only generalizes prior state-of-the-art, but also reveals subtle failure modes, explains when and why performance degrades, and offers principled solutions through temperature calibration and gating. Empirically, BAYES-ECR delivers consistent gains across 12 diverse datasets, operates effectively with both oracle and noisy LLM rankers, and runs under 1 ms per query. We believe BAYES-ECR can serve as a practical bridge between label-scarce applications and the growing availability of pairwise signals from LLMs.

REFERENCES

Alice EA Allen and Alexandre Tkatchenko. Machine learning of material properties: Predictive and interpretable multilinear models. *Science advances*, 8(18):eabm7185, 2022.

Anthropic. Anthropic api guide. `https://docs.anthropic.com/en/api/overview`, 2025. Accessed: 2025-09-16.

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

Manuela Cattelan. Models for paired comparison data: A review with emphasis on dependent data. *Statistical Science*, pp. 412–433, 2012.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30, 2017.

William G Cochran and Sarah Porter Carroll. A sampling investigation of the efficiency of weighting inversely as the estimated variance. *Biometrics*, 9(4):447–459, 1953.

Paulo Cortez. Student Performance. UCI Machine Learning Repository, 2014.

Li Di, Christopher Keefer, Dennis O Scott, Timothy J Strelevitz, George Chang, Yi-An Bi, Yurong Lai, Jonathon Duckworth, Katherine Fenner, Matthew D Troutman, et al. Mechanistic insights from comparing intrinsic clearance values between human liver microsomes and hepatocytes to guide drug design. *European journal of medicinal chemistry*, 57:441–448, 2012.

Google. Gemini developer api. `https://ai.google.dev/gemini-api/docs`, 2025. Accessed: 2025-09-16.

Taicheng Guo, Bozhao Nan, Zhenwen Liang, Zhichun Guo, Nitesh Chawla, Olaf Wiest, Xiangliang Zhang, et al. What can large language models do in chemistry? a comprehensive benchmark on eight tasks. *Advances in Neural Information Processing Systems*, 36:59662–59688, 2023.

Tin Kam Ho. Random decision forests. In *Proceedings of 3rd international conference on document analysis and recognition*, volume 1, pp. 278–282. IEEE, 1995.

Eva JI Hoeijmakers, Bibi Martens, Babs MF Hendriks, Casper Mihl, Razvan L Miclea, Walter H Backes, Joachim E Wildberger, Frank M Zijta, Hester A Gietema, Patricia J Nelemans, et al. How subjective ct image quality assessment becomes surprisingly reliable: pairwise comparisons instead of likert scale. *European Radiology*, 34(7):4494–4503, 2024.

Kexin Huang, Tianfan Fu, Wenhao Gao, Yue Zhao, Yusuf Roohani, Jure Leskovec, Connor Coley, Cao Xiao, Jimeng Sun, and Marinka Zitnik. Therapeutics data commons: Machine learning datasets and tasks for drug discovery and development. *Advances in neural information processing systems*, 2021.

Qiaohao Liang, Aldair E Gongora, Zekun Ren, Armi Tiihonen, Zhe Liu, Shijing Sun, James R Deneault, Daniil Bash, Flore Mekki-Berrada, Saif A Khan, et al. Benchmarking the performance of bayesian optimization across multiple experimental materials science domains. *npj Computational Materials*, 7(1):188, 2021.

Franco Lombardo and Yankang Jing. In silico prediction of volume of distribution in humans. extensive data set and the exploration of linear and nonlinear methods coupled with molecular interaction fields descriptors. *Journal of chemical information and modeling*, 56(10):2042–2052, 2016.

Alexander Ly, Maarten Marsman, Josine Verhagen, Raoul PPP Grasman, and Eric-Jan Wagenmakers. A tutorial on fisher information. *Journal of Mathematical Psychology*, 80:40–55, 2017.

David L Mobley and J Peter Guthrie. Freesolv: a database of experimental and calculated hydration free energies, with input files. *Journal of computer-aided molecular design*, 28:711–720, 2014.

Frederick Mosteller. Remarks on the method of paired comparisons: Iii. a test of significance for paired comparisons when equal standard deviations and equal correlations are assumed. *Psychometrika*, 16(2):207–218, 1951.

Kevin P Murphy. *Probabilistic machine learning: an introduction*. MIT press, 2022.

R Scott Obach, Franco Lombardo, and Nigel J Waters. Trend analysis of a database of intravenous pharmacokinetic parameters in humans for 670 drug compounds. *Drug Metabolism and Disposition*, 36(7):1385–1405, 2008.

OpenAI. Openai api reference. https://platform.openai.com/docs/api-reference/introduction, 2025. Accessed: 2025-09-16.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35: 27730–27744, 2022.

Zhen Qin, Rolf Jagerman, Kai Hui, Honglei Zhuang, Junru Wu, Le Yan, Jiaming Shen, Tianqi Liu, Jialu Liu, Donald Metzler, et al. Large language models are effective text rankers with pairwise ranking prompting. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pp. 1504–1518, 2024.

Jennifer Routh, Sharmini Julita Paramasivam, Peter Cockcroft, Sarah Wood, John Remnant, Cornélie Westermann, Alison Reid, Patricia Pawson, Sheena Warman, Vishna Devi Nadarajah, et al. Rating and ranking preparedness characteristics important for veterinary workplace clinical training: a novel application of pairwise comparisons and the elo algorithm. *Frontiers in Medicine*, 10:1128058, 2023.

David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *nature*, 323(6088):533–536, 1986.

Benjamin Sanchez-Lengeling and Alán Aspuru-Guzik. Inverse molecular design using machine learning: Generative models for matter engineering. *Science*, 361(6400):360–365, 2018.

Adil Shamim. Cost of international education, 2025. URL https://www.kaggle.com/datasets/adilshamim8/cost-of-international-education.

Murat Cihan Sorkun, Abhishek Khetan, and Süleyman Er. Aqsoldb, a curated reference set of aqueous solubility and 2d descriptors for a diverse set of compounds. *Scientific data*, 6(1):143, 2019.

Atharva Soundankar. Smart farming sensor data for yield prediction, 2025. URL https://www.kaggle.com/datasets/atharvasoundankar/smart-farming-sensor-data-for-yield-prediction.

Michael Sun, Gang Liu, Weize Yuan, Wojciech Matusik, and Jie Chen. Foundation molecular grammar: Multi-modal foundation models induce interpretable molecular graph languages. In *International Conference on Machine Learning*. PMLR, 2025.

LL Thurstone. A law of comparative judgment. *Psychological Review*, 34(4):273–286, 1927.

Ning-Ning Wang, Jie Dong, Yin-Hua Deng, Min-Feng Zhu, Ming Wen, Zhi-Jiang Yao, Ai-Ping Lu, Jian-Bing Wang, and Dong-Sheng Cao. Adme properties evaluation in drug discovery: prediction of caco-2 cell permeability using a combination of nsga-ii and boosting. *Journal of chemical information and modeling*, 56(4):763–773, 2016.

David Weininger. Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. *Journal of chemical information and computer sciences*, 28(1):31–36, 1988.

Kevin Tirta Wijaya, Minghao Guo, Michael Sun, Hans-Peter Seidel, Wojciech Matusik, and Vahid Babaei. Two-stage pretraining for molecular property prediction in the wild. *arXiv preprint arXiv:2411.03537*, 2024.

Kevin Tirta Wijaya, Michael Sun, Minghao Guo, Hans-Peter Seidel, Wojciech Matusik, and Vahid Babaei. Post hoc regression refinement via pairwise rankings. *arXiv preprint arXiv:2508.16495*, 2025.

Tong Wu, Guandao Yang, Zhibing Li, Kai Zhang, Ziwei Liu, Leonidas Guibas, Dahua Lin, and Gordon Wetzstein. Gpt-4v (ision) is a human-aligned evaluator for text-to-3d generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 22227–22238, 2024.

Zhenqin Wu, Bharath Ramsundar, Evan N Feinberg, Joseph Gomes, Caleb Geniesse, Aneesh S Pappu, Karl Leswing, and Vijay Pande. Moleculenet: a benchmark for molecular machine learning. *Chemical science*, 9(2):513–530, 2018.

Le Yan, Zhen Qin, Honglei Zhuang, Rolf Jagerman, Xuanhui Wang, Michael Bendersky, and Harrie Oosterhuis. Consolidating ranking and relevance predictions of large language models through post-processing. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 410–423, 2024.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623, 2023.

Daniel M Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*, 2019.

A Appendix

A.1 Detailed Proofs

Lemma 3.1. *Proof.* The regressor log-likelihood contributes $-\frac{1}{2\sigma_{\text{re}}^2}(\hat{y}_0^{\text{re}} - y_0)^2$ with curvature $-1/\sigma_{\text{re}}^2 < 0$. Each BT term adds $\log s(y_0 - y_i)$ or $\log 1 - s(y_0 - y_i)$, whose second derivative is $-s(1-s) \leq 0$. Sum of concave terms and a concave prior is concave; the Gaussian term's strictly negative curvature makes the sum strictly concave.

Lemma 3.4. *Proof.* $\mathcal{L}'_{\text{BT}}(y)$ (Proposition 3.3.) is strictly decreasing, therefore, $\mathcal{L}_{\text{BT}}(y)$ is strictly concave. Setting $\mathcal{L}'_{\text{BT}}(y) = 0$ gives the equation. Since $F(y) = \sum_i u_i(y)$ is continuous and strictly increasing with $F(-\infty) = 0$, $F(\infty) = k$, there is a unique solution at the observed count m.

Let $y_1 \leq ... \leq y_k$ be the ordered reference set and suppose the true $y_0 \in (y_m, y_{m+1})$,

- if most $(y - y_i)$ are in the sigmoid saturated regime, F behaves like a hard count, so $F(y_0) \approx m$, $F(\tilde{y}_0) = m$, and $\hat{y}_0^{\text{ra}} = \tilde{y}_0 \approx y_0$ up to the resolution $(y_{m+1} - y_m)$.
- if many $(y - y_i)$ lie in the transition region, $F(y_0) \not\approx m$, $F(\tilde{y}_0) = m$, and $\hat{y}_0^{\text{ra}} = \tilde{y}_0 \not\approx y_0$. This shifts does not induce rank bias as long as $\tilde{y}_0 \in (y_m, y_{m+1})$.
- rank bias arises if the shift exits the interval, i.e., $|\tilde{y}_0 - y_0| \geq \min\{y_0 - y_m, y_{m+1} - y_0\}$, which is more likely when the number of $(y - y_i)$ that lie in the transition region is high.

Lemma 3.6. *Proof.* Let $I_{\text{reg}} = 1/\sigma_{\text{re}}^2$ be the Fisher information of the regressor term and write the total gradient and Fisher information at y as

$$\begin{aligned} g_{\text{tot}}(y) &= g_{\text{reg}}(y) + g_{\text{rank}}(y) \\ &= -\frac{1}{\sigma_{\text{re}}^2}(y - \hat{y}_0^{\text{re}}) + \sum_{i=1}^{k}(r_i - u_i(y)), \end{aligned} \tag{20}$$

$$I_{\text{tot}}(y) = I_{\text{reg}} + I_{\text{rank}}(y). \tag{21}$$

Define the local rank target for a single Newton step

$$\begin{aligned} \tilde{y}^{\text{ra}}(y) &= y + \frac{g_{\text{rank}}(y)}{I_{\text{rank}}(y)} \\ &= y + \frac{\sum_{i=1}^{k}(r_i - u_i(y))}{I_{\text{rank}}(y)}. \end{aligned} \tag{22}$$

By substituting $g_{\text{reg}}(y) = I_{\text{reg}} \cdot (\hat{y}_0^{\text{re}} - y)$ and $g_{\text{rank}}(y) = I_{\text{rank}}(y) \cdot (\tilde{y}^{\text{ra}}(y) - y)$, a single Newton step is an information-weighted average as written in Equation 12.

Lemma 3.7. *Proof.* Using $\mathbb{E}[r_i] = (1-a) + (2a-1)\mathbb{1}(y_0 > y_i)$ to model flip errors in a noisy binary ranker gives

$$\mathbb{E}[g_{\text{rank}}(y)] = \sum_{i=1}^{k}\mathbb{E}[r_i] - \sum_{i=1}^{k}u_i(y) = k\left((1-a) + (2a-1)p^* - \hat{p}(y)\right), \tag{23}$$

If $p^* \geq 1/2 \geq \hat{p}(y)$ or $p^* \leq 1/2 \leq \hat{p}(y)$, then

$$\left|\mathbb{E}[g_{\text{rank}}(y)]\right| \geq k(2a-1)|p^* - \hat{p}(y)|. \tag{24}$$

Substituting $g_{\text{rank}}(y)$ with $-1/\sigma_{\text{re}}^2(y - \hat{y}_0^{re})$, the expected Newton step is rank-dominated whenever

$$(2a-1)|p^* - \hat{p}(y)| > \frac{|y - \hat{y}_0^{\text{re}}|}{k\sigma_{\text{re}}^2}. \tag{25}$$

A.2 USE OF LARGE LANGUAGE MODELS

We use large language models (LLMs) to (i) polish writing (restructuring sentences and proofread grammars and typos), (ii) search for related works, (iii) predict pairwise rankings of molecule pairs. For writing and search, we use ChatGPT5. For pairwise ranking predictions, we use ChatGPT5 and Claude Sonnet 4 (Copilot version). The prompt that we used for pairwise ranking prediction is as follow,

```
You are an expert molecular reasoning model tasked with
predicting pairwise rankings of molecules based on a described
molecular property of interest (e.g., solubility, polarity, etc.).

You will be given json files containing the test molecules to be
compared with the reference molecules. The property of interest
is described within the json files with the key "description".

* Your Task - Follow These Steps:
1. Read the dataset's description to understand the molecular
property being ranked (e.g., "higher solubility", "lower toxicity").
- Be careful with the mesurement unit. For example, a higher number
in IC50 could mean lower toxicity.
2. For each molecule pair (test molecule vs. reference molecule):
- Use your internal knowledge to infer which molecule ranks higher
for the property.
- You may use structural patterns, substrings, atom types,
SMARTS-like features, token-level patterns, or other insights
you may have.
- You are encouraged to develop your own heuristics or scoring logic
using Python.
3. Assign a "pairwise_rank" to each pair:
1 → test molecule ranks higher than reference, meaning
test_property > reference_property
0 → test molecule ranks lower or equal to reference
Caution! Only care for the value of the property. For example,
if toxicity is measured in IC50, and test_molecule IC50 value
is greater than reference_molecule IC50 value, you should output 1.
4. Save your results as a new JSON file with similiar structure.

* You Are Allowed To:
1. Write your own Python logic.
2. Use basic Python and string-based pattern recognition
3. Think step-by-step to develop useful ranking heuristics
4. Use helper functions from ChemInformatics libraries, as long
as you do not use them to directly predict the porperty of interest
values.

* You Are NOT Allowed To:
1. Use the internet to search for the property values
2. Use cheminformatics libraries like RDKit to directly predict
the property of interest values, e.g., solubility of molecule A
for solubility datasets.
3. Access files not specified in this prompt

* Each entry in your output JSON should look like this:
"test_molecule": "smiles": "...",
"reference_molecules": [
    {   "id": "...",
        "smiles": "..."
        "pairwise_ranks": 1,
```

```
    },
that is, put the pairwise_rank predictions inside the
reference_molecule. Your output json file should be named
"pairwise_ranking_predictions_{split_id}.json"

* Tips for Better Performance:
1. Think aloud: before you begin ranking, describe what is the
property of interest and why one molecule might rank higher
based on your knowledge
2. Use token or substring patterns (e.g., "more OH groups" or
"more aromatic rings")
3. Define scoring rules: e.g., "count('O') - count('N')"
to estimate polarity
```

A.3 Effects of number of samples for temperature calibration

In Section 3.3, the temperature τ is set to $\hat{\tau}$cal, the calibrated value estimated from the labeled reference set. A natural question is the robustness of this procedure in data-scarce regimes, where only a few labeled references are available. Figure 5 shows $\hat{\tau}$cal as a function of the calibration set size k_{cal}. Across the 9 TDC ADMET datasets, $\hat{\tau}$cal converges at around $k_{\text{cal}} \approx 10$, indicating that optimal calibration can be obtained with only 10 labeled samples. Furthermore, except for PPBR_AZ, the estimates obtained with $k_{\text{cal}} < 10$ are already close to their converged values. This suggests that $\hat{\tau}$cal remains reliable even when very limited calibration data are available. Consistent with this observation, in the experiment where we fix the reference set size at $k = 50$ but vary $k_{\text{cal}} \in [3, 50]$ (Figure 6), the resulting performance curves are nearly identical. Together, these results demonstrate that the number of references used for temperature calibration has a negligible effect on overall regression enhancement performance.

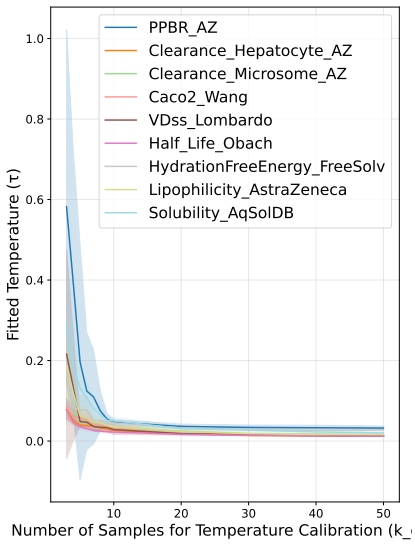

Figure 5: Effects of number of samples used to calibrate the temperature on the value of $\hat{\tau}_{\text{cal}}$.

A.4 Regression enhancement for multiple targets

The discussion and experiments in Sections 3 and 4 focus on scalar regression tasks. In many practical settings, however, the goal is to predict multiple properties simultaneously, i.e., to produce vector-valued outputs. Extending Bayes-ECR to this setting is straightforward: one can assume independence across output dimensions and apply Bayes-ECR separately to each component of a target vector $\boldsymbol{y} \in \mathbb{R}^d$, yielding d independent regression enhancement processes.

In practice, the components of $\boldsymbol{y}$ are often correlated. For example, in molecular chemistry, aqueous solubility and membrane permeability are often related, as are clearance and half-life. In such cases, dependencies among outputs can be captured by employing a low-rank approximation of the covariance structure, which enables a computationally efficient implementation. Exploring this extension is an interesting direction for future work.

A.5 More Results on TDC ADMET

We show results for more k values in Figure 7-Figure 15. In general, Bayes-ECR can improve the regression error (i.e., $\beta < 1$) with a reference set size as small as $k = 3$.

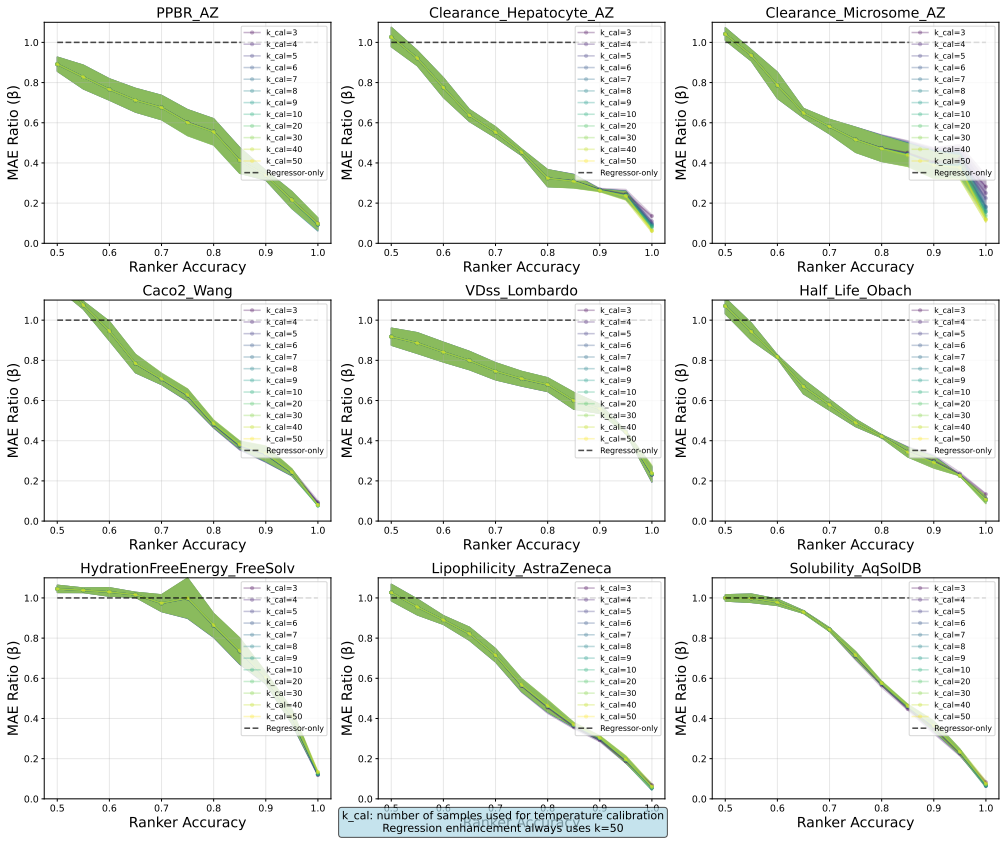

Figure 6: Effects of number of samples on the regression enhancement. We measure β (lower is better) as a function of number of samples used to calibrate the temperature k_{cal}. The reference set size k is set to 50.

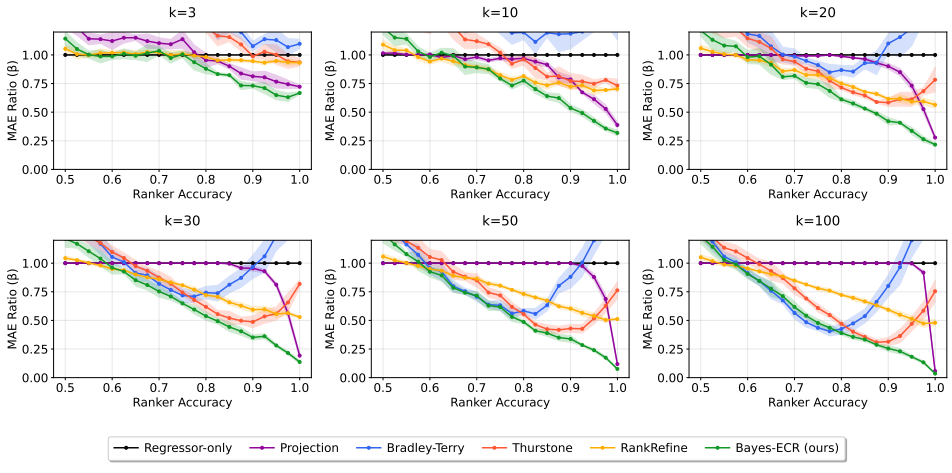

Figure 7: Caco2_Wang

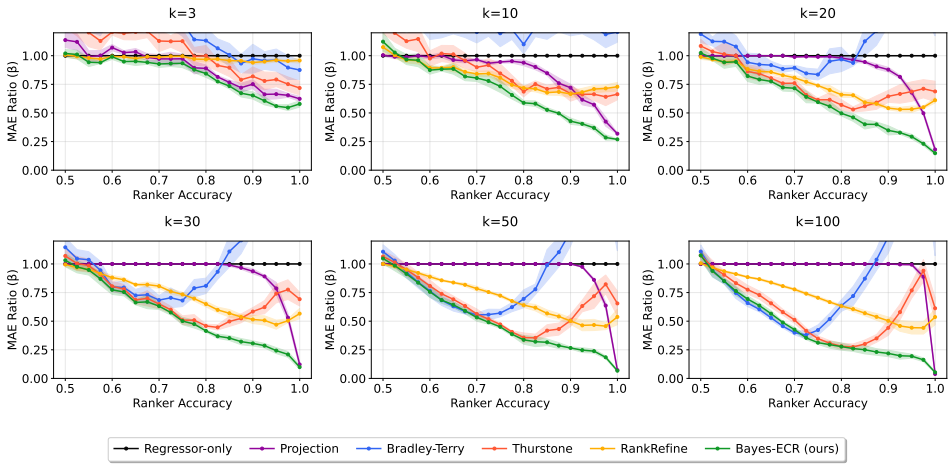

Figure 8: Clearance_Hepatocyte_AZ

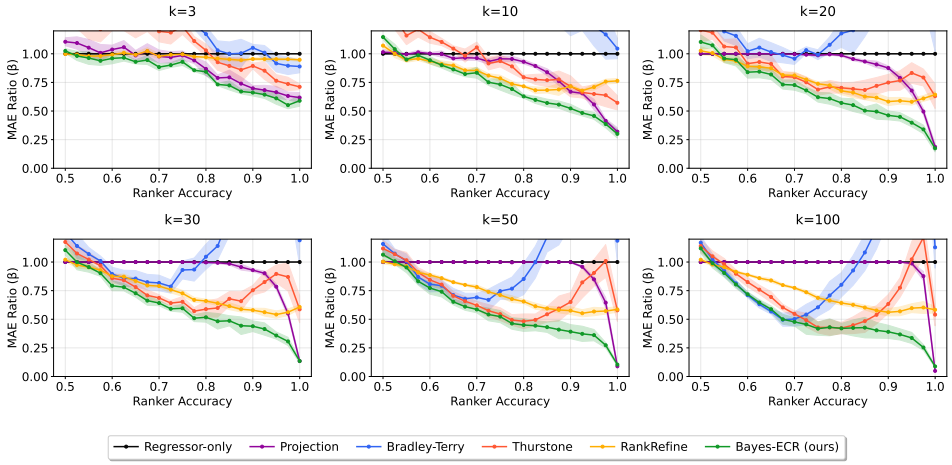

Figure 9: Clearance_Microsome_AZ

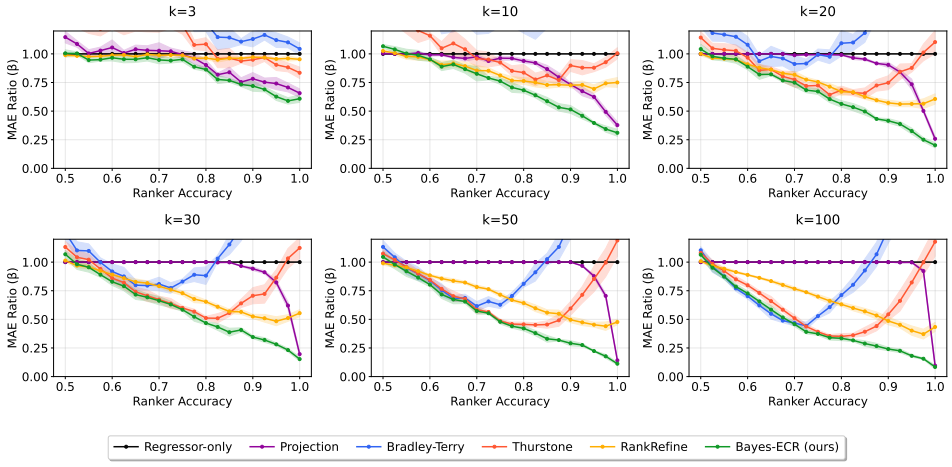

Figure 10: Half_Life_Obach

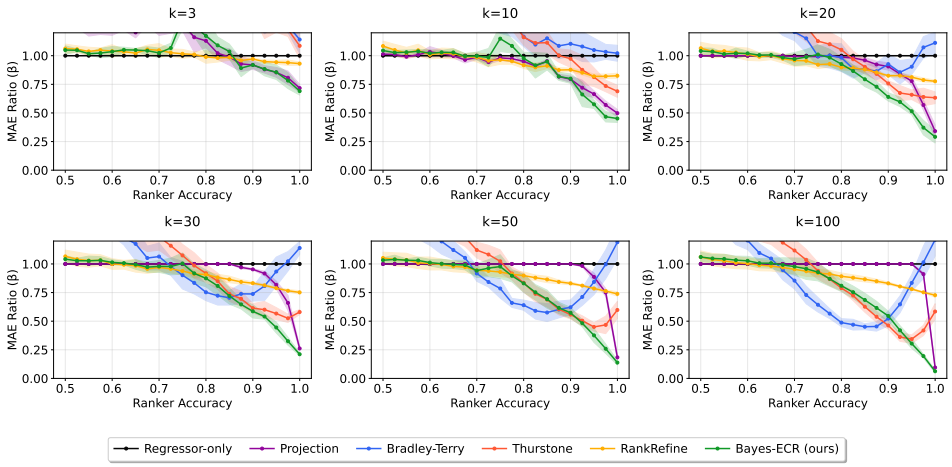

Figure 11: HydrationFreeEnergy_FreeSolv

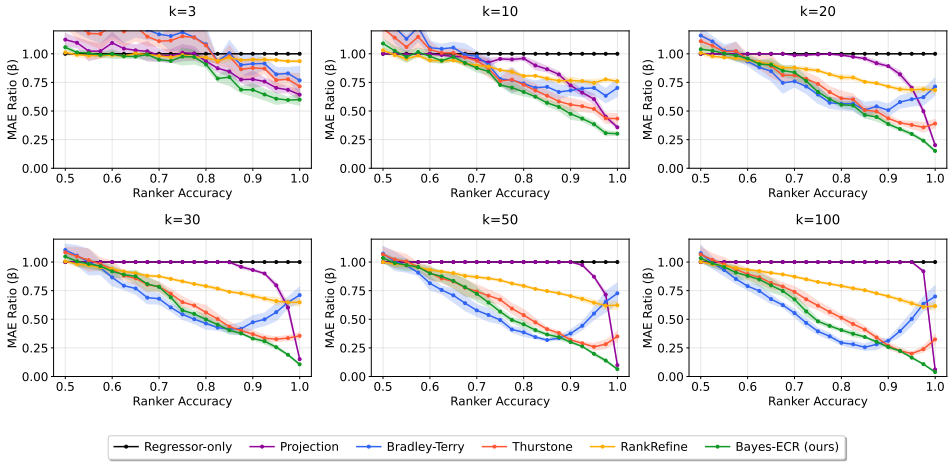

Figure 12: Lipophilicity_AstraZeneca

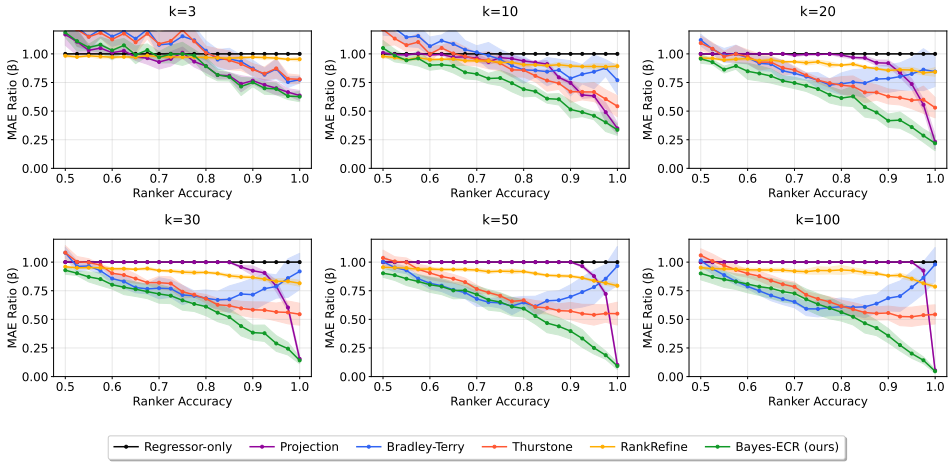

Figure 13: PPBR_AZ

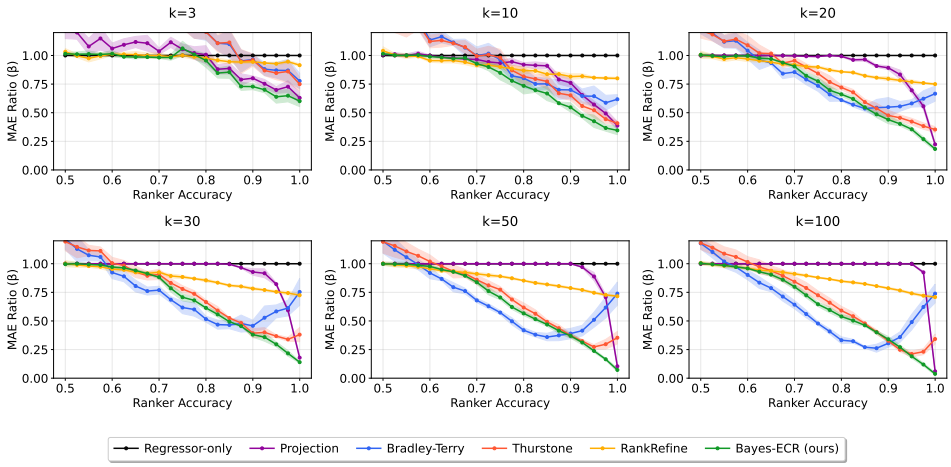

Figure 14: Solubility_AqSolDB

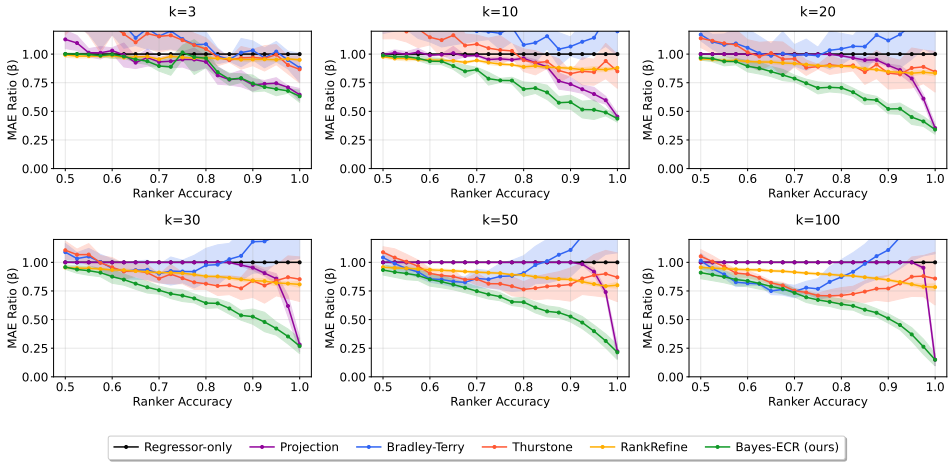

Figure 15: VDss_Lombardo

A.6 REBUTTAL

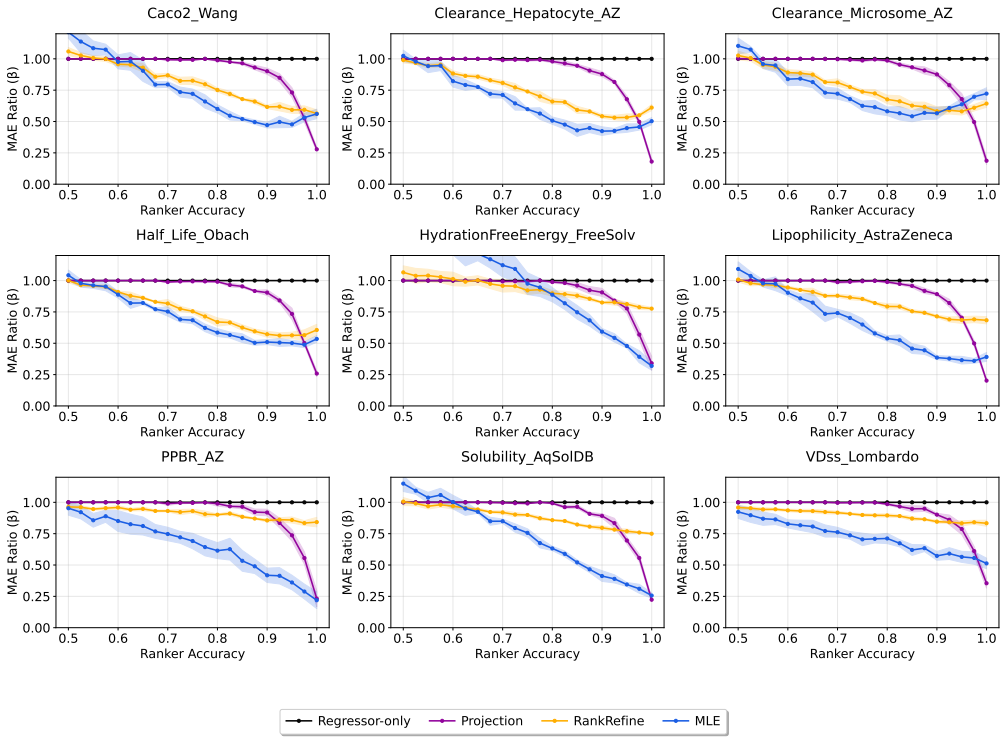

Figure 16: Non-calibrated Bayes-ECR vs. RankRefine and Projection. Reference set size k = 20. Non-calibrated Bayes-ECR consistently achieves lower error compared to RankRefine and Projection baselines across datasets, often by a large margin

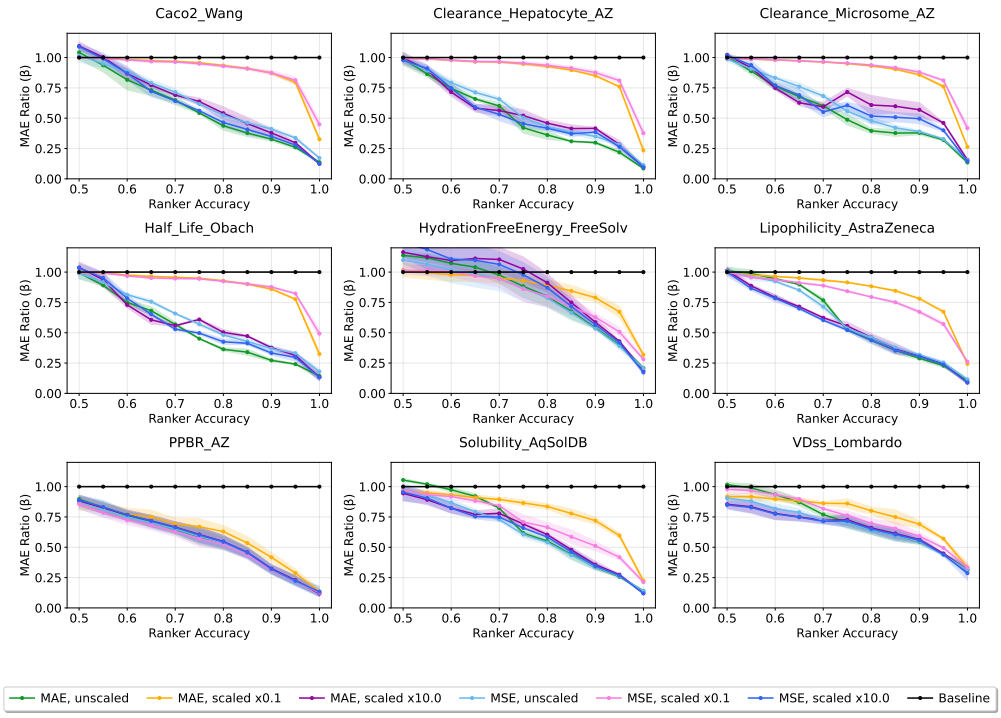

Figure 17: Sensitivity analysis on under or overestimation of regressor uncertainty. Overestimating uncertainty (e.g., s = 10.0) has minimal impact on performance, while underestimating it (e.g., s = 0.1) degrades performance due to excessive confidence in the regressor.

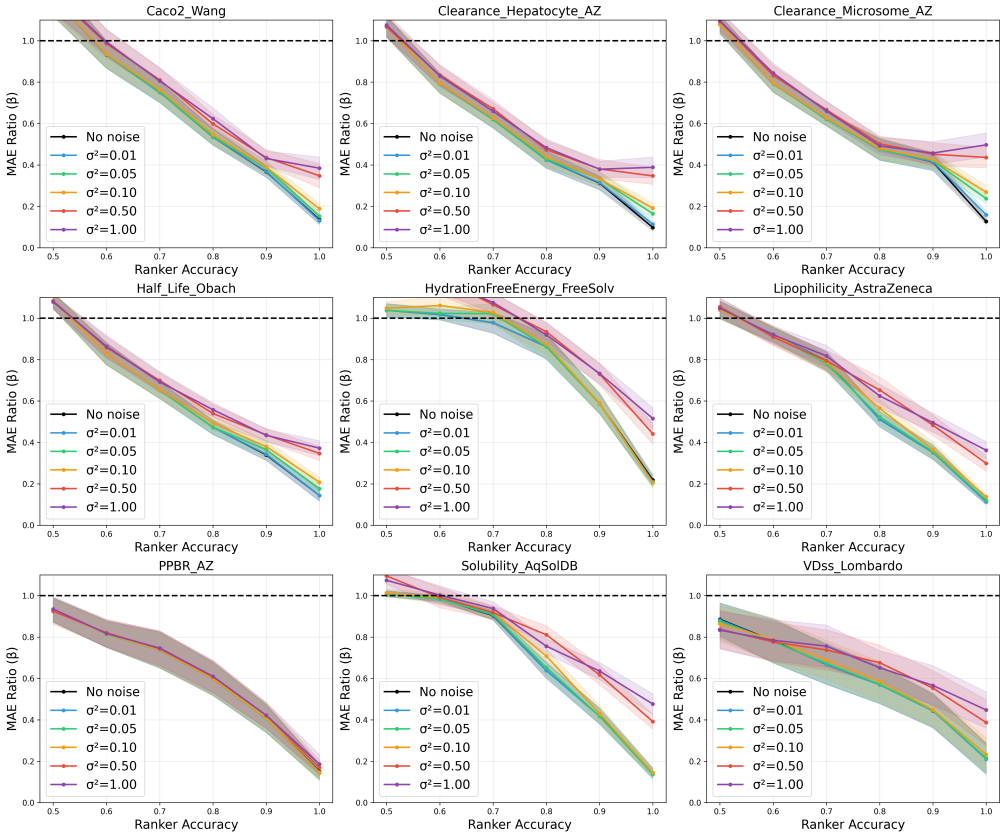

Figure 18: Sensitivity analysis when ranker accuracy is under or overestimated, resulting in non-optimal tau values. Bayes-ECR is robust even when using a suboptimal tau value caused by softgating bias

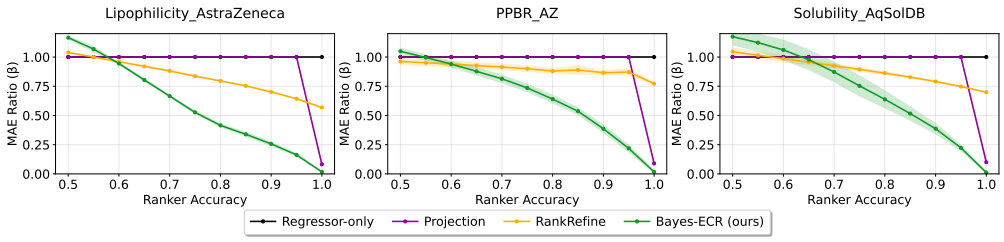

Figure 19: We evaluated our method, Bayes-ECR, on three larger ($\geq$1000 training labels) TDC ADME datasets: Lipophilicity, PPBR, and Solubility. With a reference set size of 1000, Bayes-ECR consistently delivered the best overall performance across the entire spectrum of ranker accuracy.

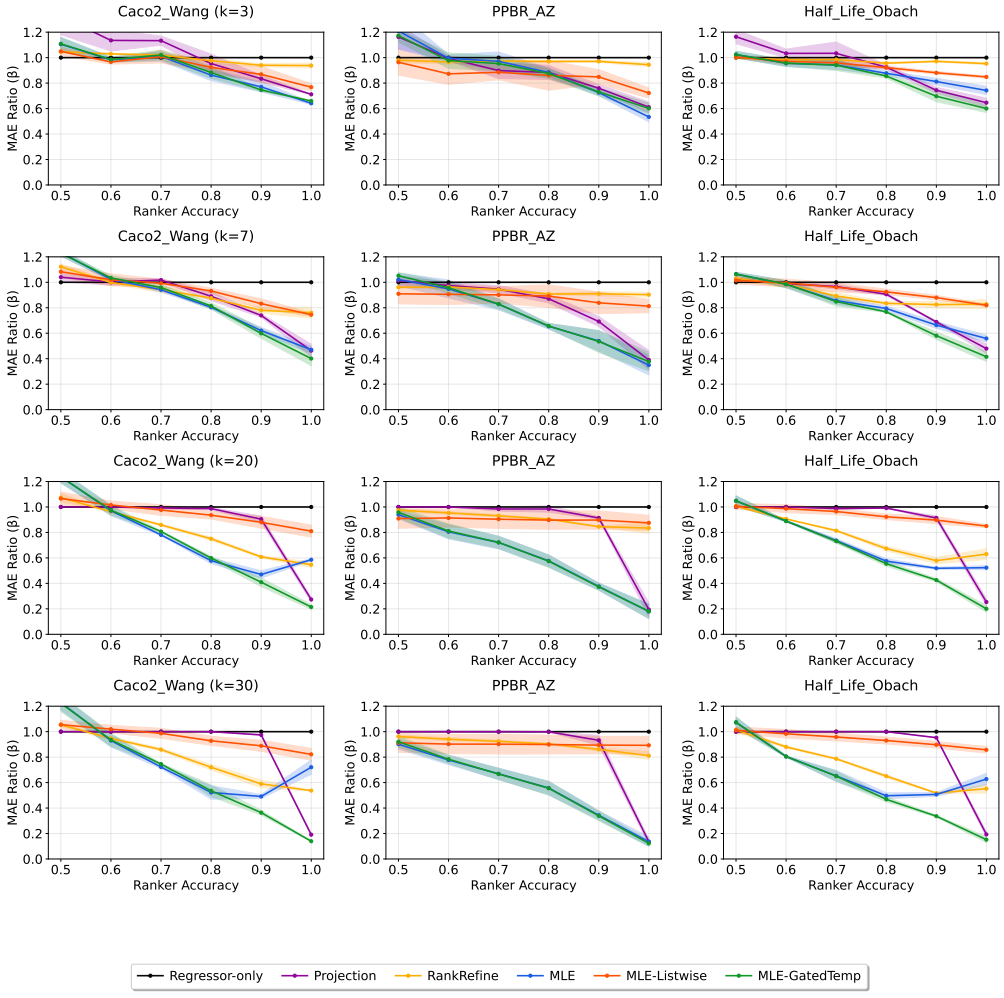

Figure 20: We extend our method, Bayes-ECR, to listwise rankings (MLE-Listwise). For $k = 3$, the listwise variant achieves reasonable performance, confirming that Bayes-ECR can indeed be extended to listwise ranking. However, its performance is not yet on par with our pairwise version.