# Prediction-Powered Active Testing

**Kianoosh Ashouritaklimi**[*] University of Oxford
**Valentin Kilian** University of Oxford
**Daolang Huang** Aalto University
**Tom Rainforth** University of Oxford
**François Caron** University of Oxford

## Abstract

Active testing provides a label-efficient approach to risk estimation by adaptively selecting which test points should be labelled. However, existing estimators fail to exploit the informative predictions of powerful black-box models, even though such predictions are increasingly available in settings where labels remain expensive. To address this, we propose **Prediction-Powered Active Testing (PPAT)**, a novel label-efficient risk estimation framework that combines the unbiased LURE estimator [22] with a prediction-powered control variate. Rather than using proxy predictions as biased pseudo-labels, PPAT uses them to residualise the loss, preserving unbiasedness while reducing variance. Beyond the estimator itself, PPAT also changes which points should be acquired: we derive oracle and practical surrogate-based acquisition rules tailored to reducing the variance of our estimator. Moreover, we establish asymptotic normality for PPAT, yielding asymptotically valid confidence intervals and thus a principled estimate of the uncertainty around our estimates. Across tabular regression and image-classification tasks, PPAT outperforms existing methods in risk estimation, while its confidence intervals attain the target coverage with substantially fewer labels and smaller widths.

## 1 Introduction

Active learning [5, 7, 45, 69] reduces the cost of training, but in many applications the cost of *evaluation* is just as important. In many domains, large unlabelled test pools are often easy to obtain, while ground-truth labels remain expensive and require expert annotation. Active testing [10, 36, 39, 40, 67, 68, 81, 82] addresses this bottleneck by adaptively sampling which test points to label while correcting for the bias introduced by adaptive sampling. Crucially, adaptive sampling can achieve lower variance than random sampling under suitable acquisition strategies [22], while remaining unbiased.

In many settings, such as image classification [34, 44, 61, 83], language modelling [13, 72, 74, 78], and biomolecular structure prediction [1, 16, 35, 60], we also have access to a powerful black-box predictor that can produce cheap predictions on the entire test pool. A natural question here is: *can we leverage these predictions to produce more accurate, label-efficient estimates of our risk?* One approach would be to replace the true, unknown labels with the predictions of the black-box model, similar to the work of [40]. This, however, generally results in biased estimates of the risk. While it can yield accurate estimates of the risk when the labelling budget is very small and the black-box predictor is already highly accurate, the resulting estimator is generally biased, with the bias becoming increasingly problematic as the labelling budget grows.

To address this, we propose **Prediction-Powered Active Testing (PPAT)**, which combines the unbiased importance-weighted LURE estimator [22] with a prediction-powered control variate built from cheap black-box predictions [3]. Rather than using these predictions as biased pseudo-labels,

[*]Correspondence to: ashouritaklimi@stats.ox.ac.uk

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

PPAT uses them to *residualise* the loss: it subtracts, from each true loss, the loss induced by the black-box prediction after centring it by its average over the test pool, thereby preserving the unbiasedness of active testing. Our approach introduces a tuning parameter $\lambda$ that controls the strength of this correction and recovers LURE as the special case $\lambda = 0$, making PPAT a principled extension of the LURE estimator. We show that this control-variate formulation can reduce variance relative to LURE and discuss practical choices of $\lambda$.

Beyond the estimator itself, PPAT also changes what should be sampled. Rather than reusing active-testing proposals designed for the raw loss, we derive proposals tailored to the residualised objective induced by our estimator. This leads to an oracle proposal and a practical surrogate acquisition rule that samples points expected to be most informative for reducing the variance of the resulting risk estimate. We also establish asymptotic normality for the PPAT estimator, which allows us to construct asymptotically valid confidence intervals. Empirically, we evaluate PPAT extensively on tabular regression and image-classification tasks. We find that PPAT outperforms existing methods in risk estimation, while also attaining the target coverage level with substantially fewer labels and narrower confidence intervals.

## 2 Background

Throughout this paper, we reserve capital letters for random variables and lowercase letters for observations. For readability, we state the results in the main body using the simplified notation introduced below; their formal versions are given in §E.

Let $\mathcal{D} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$ be an unlabelled test pool of size $N$, where each input $\mathbf{x}_i \in \mathcal{X}$ has a true, unknown label $y_i \in \mathcal{Y}$. We treat both the inputs and labels as fixed quantities. Furthermore, let $f : \mathcal{X} \to \mathcal{Y}$ be a fixed function we wish to evaluate for a given loss function $\mathcal{L} : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$. We define the loss for each test data point as $\ell_i := \mathcal{L}(f(\mathbf{x}_i), y_i)$ for $i = 1, \dots, N$, and the sequence of these losses as $\ell_{1:N} := (\ell_1, \dots, \ell_N)$. For a fixed labelling budget $M \ll N$, our objective[2] is to estimate the test pool risk

$$R = \frac{1}{N} \sum_{i=1}^{N} \ell_i \tag{1}$$

based on the inputs $\mathcal{D}$ and on $M$ queried labels $y_{I_1}, \dots, y_{I_M}$, with indices $I_1, \dots, I_M$. A naive approach is to sample the indices randomly without replacement from $[N] := \{1, \dots, N\}$ and use the unweighted estimator

$$\widehat{R}_{\text{unw}} = \frac{1}{M} \sum_{m=1}^{M} \ell_{I_m}. \tag{2}$$

This estimator is unbiased but typically suffers from high variance [22, 39].

### 2.1 Active Testing and the LURE Estimator

Active testing (AT) [39] aims to estimate the test pool risk $R$ more efficiently by sequentially selecting the most useful points to label. More formally, for each acquisition round $m$, let $S_m \subseteq \{1, \dots, N\}$ denote the remaining indices to be sampled and $\mathcal{F}_{m-1}$ the $\sigma$-algebra generated by all the information available before round $m$ (e.g. $D$, past queried indices and labels). AT selects the $m$th index to query by sampling from a proposal $Q_m$:

$$Q_m(\cdot) := \mathbb{P}(I_m = \cdot \mid \mathcal{F}_{m-1}) \quad \text{supported on } S_m. \tag{3}$$

To estimate the risk, AT uses the LURE (*Levelled Unbiased Risk Estimator*) estimator of [22]:

$$\widehat{R}_{\text{LURE}} = \frac{1}{M} \sum_{m=1}^{M} V_m \ell_{I_m}, \tag{4}$$

where $I_m \mid I_{1:m-1} \sim Q_m$ and

$$V_m = 1 + \frac{N - M}{N - m} \left( \frac{1}{(N - m + 1)\, Q_m(I_m)} - 1 \right). \tag{5}$$

[2]We target the test pool risk instead of the population risk, $R_P = \mathbb{E}_{(X,Y)\sim P}[L(f(X), Y)]$, as in practice we only observe one test pool and inference about $R_P$ would require additional assumptions linking our pool to $P$.

This is an *unbiased* estimate of $R$ and, when $\mathcal{L}$ is nonnegative, has strictly *lower variance* than random sampling without replacement under the oracle proposal $Q_m^\star(i) \propto \ell_i$ [22, Thm. 5]. For general real-valued losses, the variance is instead minimised greedily at each step using the oracle *myopic* proposal $Q_m^\star(i) \propto |\ell_i|$ (see §E.4 for further details). In practice, the true labels are not known, and so $Q_m^\star$ is approximated[3] with

$$Q_m^{\mathrm{AT}}(i) \;\propto\; \sqrt{\mathbb{E}_{\pi_m(Y|\mathbf{x}_i)}[\,\mathcal{L}(f(\mathbf{x}_i), Y)^2\,]}, \tag{6}$$

where $\pi_m(Y \mid \mathbf{x})$ is a *surrogate model* used to approximate $p(Y \mid \mathbf{x})$ at round $m$ [39].

### 2.2 Active Surrogate Estimators

*Active Surrogate Estimators* (ASEs, [40]) provide an alternative approach to AT that replaces importance-weighted Monte Carlo estimation with an interpolation-based surrogate estimate of the loss. Rather than estimating the risk directly from the queried losses, ASEs learn a surrogate model for the unknown test-time label distribution and use it to impute losses across the full test pool. Concretely, after $M$ acquisitions, the estimator is

$$\widehat{R}_{\mathrm{ASE}} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{E}_{Y\sim\pi_M(\cdot|\mathbf{x}_i)}[\mathcal{L}(f(\mathbf{x}_i), Y)]\,.$$

In contrast to LURE-based AT, the acquisition in ASE is designed to improve the surrogate itself rather than to produce an unbiased importance-weighted estimator. To this end, [40] proposed the *Expected Weighted Disagreement (XWED)* acquisition function, which prefers points with high epistemic uncertainty under $\pi_m$ that are also expected to contribute strongly to the final risk estimate (see §D.3.1 for additional details).

A key advantage of ASEs is that they can make more effective use of surrogate generalisation and allow for more flexible acquisition strategies, including deterministic ones. However, they do not retain the finite-sample unbiasedness of LURE: surrogate misspecification and finite-sample estimation error enter the final estimator directly. Consequently, their performance depends strongly on the quality of the learned surrogate and they suffer from weaker theoretical guarantees than LURE-based AT.

### 2.3 Control-Variate Estimator

We conclude this background section with a short review of control variate estimators. Let $\widehat{R}$ be an unbiased estimator of $R$. Let $\widehat{C}$ be a zero-mean random variable, called the *control variable* [24]. For $\lambda \in \mathbb{R}$, the control variate estimator, $\widehat{R}_{\mathrm{cv}}$, of $R$ is defined as

$$\widehat{R}_{\mathrm{cv}} = \widehat{R} - \lambda\widehat{C}. \tag{7}$$

The centred variable $\widehat{C}$ acts as a control-variate correction to the estimator $\widehat{R}$. By design, $\widehat{R}_{\mathrm{cv}}$ is unbiased if $\widehat{R}$ is, with variance $\mathrm{Var}(\widehat{R}_{\mathrm{cv}}) = \mathrm{Var}(\widehat{R}) - 2\lambda\mathrm{Cov}(\widehat{R},\widehat{C}) + \lambda^2\mathrm{Var}(\widehat{C})$. Moreover, $\widehat{R}_{\mathrm{cv}}$ has lower variance than $\widehat{R}$ whenever $\lambda \in (\min(0, 2\lambda^*), \max(0, 2\lambda^*))$, where $\lambda^* = \mathrm{Cov}(\widehat{R},\widehat{C})/\mathrm{Var}(\widehat{C})$ results in the largest variance reduction. At $\lambda^\star$, $\mathrm{Var}(\widehat{R}_{\mathrm{cv}}) = (1-\rho^2)\mathrm{Var}(\widehat{R})$, where $\rho$ is the correlation between $\widehat{R}$ and $\widehat{C}$. Thus, a larger absolute correlation implies a larger variance reduction.

## 3 Prediction-Powered Active Testing

We now describe our approach, **Prediction-Powered Active Testing (PPAT)**, which leverages a cheap proxy labeller $g$ to reduce the variance of the LURE estimator in a similar spirit to [3, 4].

Assume we can evaluate a cheap proxy labeller $g : \mathcal{X} \to \mathcal{Y}$ on all test inputs, yielding proxy labels $\widetilde{y}_i = g(\mathbf{x}_i)$. Define the *proxy loss* $\widetilde{\ell}_i := \mathcal{L}(f(\mathbf{x}_i), \widetilde{y}_i)$ and the associated proxy size-$N$ test risk

[3] Note that when the true loss is unknown, the oracle $Q_m^\star \propto |\ell_i|$ is better approximated by the root-mean-square score (6) rather than by an expected absolute value; see §D.1 for more details.

$\widetilde{R} := \frac{1}{N}\sum_{i=1}^{N}\widetilde{\ell}_i$. We first note that, for any $\lambda \in \mathbb{R}$, the test risk can be written alternatively as

$$R = \frac{1}{N}\sum_{i=1}^{N}(\ell_i - \lambda(\widetilde{\ell}_i - \widetilde{R})), \tag{8}$$

that is, an average of the *residualised losses* $(\ell_i - \lambda(\widetilde{\ell}_i - \widetilde{R}))_{i=1}^{N}$. We propose to apply the LURE weighting scheme to this residualised representation of $R$, yielding our PPAT estimator:

$$\widehat{R}_{\mathrm{PPAT}}(\lambda) = \frac{1}{M}\sum_{m=1}^{M} V_m(\ell_{I_m} - \lambda(\widetilde{\ell}_{I_m} - \widetilde{R})). \tag{9}$$

Our estimator recovers the standard LURE estimator (4) as a special case when $\lambda = 0$. Moreover, noting that

$$\widehat{R}_{\mathrm{PPAT}}(\lambda) = \widehat{R}_{\mathrm{LURE}} - \lambda\widehat{C}_{\mathrm{LURE}},$$

the PPAT estimator can be interpreted as a control-variate correction of $\widehat{R}_{\mathrm{LURE}}$, with the control variate being

$$\widehat{C}_{\mathrm{LURE}} = \frac{1}{M}\sum_{m=1}^{M} V_m(\widetilde{\ell}_{I_m} - \widetilde{R}),$$

where $\widehat{C}_{\mathrm{LURE}}$ is itself a zero-mean LURE-style estimator[4]. This control-variate view suggests that suitable choices of $\lambda$ should reduce variance relative to LURE. We make this more precise in the next section, where we study the finite-sample properties of the PPAT estimator.

### 3.1 Finite-Sample Properties of PPAT

The results below hold for any $1 \le M < N$, and any choice of proposals $Q_m$. We first note that our estimator naturally inherits the unbiasedness of LURE.

**Proposition 3.1** (Unbiasedness)**.** *For any fixed $\lambda \in \mathbb{R}$, we have $\mathbb{E}\Big[\widehat{R}_{\mathrm{PPAT}}(\lambda)\Big] = R$.*

*Proof.* See §E.3.

Secondly, as $\widehat{R}_{\mathrm{PPAT}}(\lambda)$ is a control-variate estimator, it follows that it achieves lower variance than $\widehat{R}_{\mathrm{LURE}}$ for a range of $\lambda$ values.

**Proposition 3.2** (Variance of $\widehat{R}_{\mathrm{PPAT}}$)**.** *Let $\lambda \in \mathbb{R}$. We have*

$$\mathrm{Var}\Big(\widehat{R}_{\mathrm{PPAT}}(\lambda)\Big) = \mathrm{Var}(\widehat{R}_{\mathrm{LURE}}) - 2\lambda\,\mathrm{Cov}(\widehat{R}_{\mathrm{LURE}}, \widehat{C}_{\mathrm{LURE}}) + \lambda^2\mathrm{Var}(\widehat{C}_{\mathrm{LURE}}). \tag{10}$$

*Thus, $\mathrm{Var}\Big(\widehat{R}_{\mathrm{PPAT}}(\lambda)\Big) < \mathrm{Var}(\widehat{R}_{\mathrm{LURE}})$ whenever $\lambda \in (\min\{0, 2\lambda^\star\}, \max\{0, 2\lambda^\star\})$ and $\mathrm{Var}(\widehat{C}_{\mathrm{LURE}}) > 0$, where*

$$\lambda^\star = \frac{\mathrm{Cov}(\widehat{R}_{\mathrm{LURE}}, \widehat{C}_{\mathrm{LURE}})}{\mathrm{Var}(\widehat{C}_{\mathrm{LURE}})}. \tag{11}$$

*In particular, $\lambda^\star$ minimises the variance of $\widehat{R}_{\mathrm{PPAT}}(\lambda)$.*

*Proof.* See §E.3.

Note that under $\lambda^*$, the variance of the PPAT estimator can also be written as $\mathrm{Var}(\widehat{R}_{\mathrm{LURE}})(1-\rho^2)$, where $\rho$ is the correlation between $\widehat{R}_{\mathrm{LURE}}$ and $\widehat{C}_{\mathrm{LURE}}$. Here, a larger correlation between these two terms results in a larger reduction in variance relative to LURE.

While Prop. 3.2 shows that PPAT can achieve lower variance than LURE, this depends critically on our choice of $\lambda$ and holds only when both LURE and PPAT use the same proposals (see also §C.5). Moreover, although it is possible to adopt the same LURE proposal, $Q_m^{AT}$, for our approach and tune only $\lambda$, this will typically be suboptimal as it selects points by their raw loss $\ell_i$, whereas the variance of PPAT is driven by the residual $\ell_i - \lambda(\widetilde{\ell}_i - \widetilde{R})$. In the following sections, we derive a proposal tailored to our PPAT estimator and discuss practical choices of $\lambda$.

[4]Indeed, we have that $\mathbb{E}[\widehat{C}_{\mathrm{LURE}}] = \frac{1}{N}\sum_{i=1}^{N}(\tilde{\ell}_i - \tilde{R}) = 0$.

### 3.2 PPAT Proposal

While the previous results hold for any valid proposal $Q_m$, we would ideally like to choose the proposal that minimises the variance of the PPAT estimator. As the exact finite-horizon variance-minimising proposal is intractable, we instead consider proposals that minimise the alternative, myopic proxy described below.

Firstly, note that since $\widehat{R}_{\mathrm{PPAT}}$ is itself a LURE estimator applied to the residualised losses $\ell_i - \lambda(\widetilde{\ell}_i - \widetilde{R})$, its sampling variance decomposes across rounds in the same way as that of LURE (see §E.4 for further details). In particular, conditionally on $I_{1:m-1}$, the $m$th contribution to this variance is proportional to

$$\mathrm{Var}\left( \frac{\ell_{I_m} - \lambda(\widetilde{\ell}_{I_m} - \widetilde{R})}{N\, Q_m(I_m)} \;\middle|\; I_{1:m-1} \right). \tag{12}$$

Thus, at round $m$, a natural myopic oracle is the proposal which minimises this conditional variance over the remaining pool $S_m$. This oracle admits a simple closed form.

**Proposition 3.3** (Myopic oracle proposal)**.** *Fix $\lambda \in \mathbb{R}$. For each round $m$, among all proposals supported on $S_m$, the myopic proxy* (12) *is minimised by*

$$Q^{\star}_{m,\lambda}(i) = \frac{|\ell_i - \lambda(\widetilde{\ell}_i - \widetilde{R})|}{\sum_{j \in S_m} |\ell_j - \lambda(\widetilde{\ell}_j - \widetilde{R})|}.$$

*Proof.* See §E.4.

In practice, this oracle proposal cannot be used, as it depends on the unknown labels $y_i$ through the true losses $\ell_i$. As in §2.1, we approximate the oracle using the surrogate predictive distribution $\pi_m(\cdot \mid \mathbf{x}_i)$, giving the surrogate score[5]

$$a_{m,\lambda}(i) := \sqrt{\mathbb{E}_{Y \sim \pi_m(\cdot \mid \mathbf{x}_i)} \left[ \Big( \mathcal{L}(f(\mathbf{x}_i), Y) - \lambda(\widetilde{\ell}_i - \widetilde{R}) \Big)^2 \right]}, \tag{13}$$

and the PPAT proposal

$$Q^{\mathrm{PPAT}}_{m,\lambda}(i) = \frac{a_{m,\lambda}(i)}{\sum_{j \in S_m} a_{m,\lambda}(j)}, \qquad i \in S_m. \tag{14}$$

To ensure strict positivity, this proposal is mixed with a uniform distribution over $S_m$ like in [39], replacing $Q^{\mathrm{PPAT}}_{m,\lambda}(i)$ by $(1-\varepsilon)\, Q^{\mathrm{PPAT}}_{m,\lambda}(i) + \varepsilon/|S_m|$ for a small $\varepsilon \in (0,1)$.

### 3.3 Choosing $\lambda$

As mentioned previously, the variance reduction achieved by PPAT depends critically on the choice of $\lambda$. More specifically, Prop. 3.2 shows that PPAT achieves lower variance than LURE only for $\lambda \in (\min\{0, 2\lambda^\star\}, \max\{0, 2\lambda^\star\})$; a $\lambda$ with the wrong sign, or with too large a magnitude, lies outside this range and instead inflates the variance relative to LURE. Selecting $\lambda$ appropriately is therefore crucial, but is also further complicated by the fact that it affects both the PPAT estimator *and* the PPAT proposal. Indeed, for given proposals $Q_m$, the control-variate coefficient $\lambda^\star$ in (11) depends on the sampling law they induce, which cannot be estimated from a single active-testing run. Moreover, $\lambda$ also parameterises our proposal, $Q^{\mathrm{PPAT}}_{m,\lambda}$, and so shapes the sampling law on which $\lambda^\star$ itself depends.

These considerations motivate two practical strategies for choosing $\lambda$: a *heuristic* choice of fixed values, and an *online* estimate updated from the labels acquired during testing.

**Heuristic, fixed choices of $\lambda$.** In a similar spirit to [3], we propose the fixed values $\lambda = 1$ and $\lambda = \frac{1}{2}$, where we expect $\lambda = 1$ to be most effective when the proxy is strongly predictive of the true loss, and use $\lambda = \frac{1}{2}$ as a more conservative choice. To see why these are sensible, first note that $\lambda^\star$

[5] As in §2.1, the root-mean-square score (13) – rather than an expected absolute value – is the more appropriate surrogate of the oracle here.

can be written as $\lambda^\star = \rho\,\sigma_R/\sigma_C$, where $\rho$ is the correlation between $\widehat{R}_{\mathrm{LURE}}$ and $\widehat{C}_{\mathrm{LURE}}$ and $\sigma_R, \sigma_C$ are their standard deviations. When the proxy losses vary on a scale comparable to the true losses, we have $\sigma_R \approx \sigma_C$, so that $\lambda^\star \approx \rho$. Moreover, for an informative proxy whose loss tracks the true loss, we expect $\rho > 0$. Under these conditions, PPAT reduces variance over LURE when $\lambda \in (0, 2\rho)$ and, consequently, $\lambda = 1$ yields a reduction when $\rho > \frac{1}{2}$, while $\lambda = \frac{1}{2}$ does so over the wider range $\rho > \frac{1}{4}$. For a genuinely useful proxy, we expect it to be strongly correlated with the true loss, giving $\rho > \frac{1}{2}$, in which case both choices reduce variance; a less useful proxy, with only moderate correlation $\frac{1}{4} < \rho \leq \frac{1}{2}$, falls outside the range for $\lambda = 1$ but is still handled by the more conservative $\lambda = \frac{1}{2}$.

**Online estimation of $\lambda$.** As $\lambda^\star$ depends on the sampling law induced by the proposals $Q_m$, it cannot be estimated from a single active-testing run. We therefore target an alternative coefficient $\lambda^\dagger$ that (i) does not depend on the proposal, and so can be estimated online from the acquired labels, and (ii) flattens the residualised losses $\ell_i - \lambda c_i$ across the pool, which minimises an upper bound on the estimator's variance. Concretely, writing $c_i = \widetilde{\ell}_i - \widetilde{R}$, we define

$$\lambda^\dagger := \arg\min_{\lambda \in \mathbb{R}} \frac{1}{N} \sum_{i=1}^{N} (\ell_i - \lambda c_i)^2 = \frac{N^{-1} \sum_{i=1}^{N} \ell_i c_i}{N^{-1} \sum_{i=1}^{N} c_i^2},$$

which is well-defined whenever $N^{-1} \sum_{i=1}^{N} c_i^2 > 0$. Here, $\lambda^\dagger$ can be seen as choosing the amount of proxy correction to subtract so that the residualised losses are as flat as possible across the pool. Moreover, under mild assumptions about our proposal, it can be shown that this minimises an upper bound on the variance of our estimator:

**Proposition 3.4** (Variance bound for PPAT)**.** *Suppose that our proposals satisfy a uniform overlap condition: there exists $\beta > 0$ such that*

$$Q_m(i) \geq \frac{\beta}{N}, \qquad m = 1, \ldots, M, \;\; i \in S_m.$$

*Then, for every $\lambda \in \mathbb{R}$,*

$$\mathrm{Var}\big(\widehat{R}_M^{\mathrm{PPAT}}(\lambda)\big) \;\leq\; \frac{1}{\beta M} \, \frac{N}{N - M + 1} \left( \frac{1}{N} \sum_{i=1}^{N} (\ell_i - \lambda c_i)^2 \right).$$

*Moreover, whenever $N^{-1} \sum_{i=1}^{N} c_i^2 > 0$, this bound is minimised over $\lambda$ by*

$$\lambda^\dagger = \frac{N^{-1} \sum_{i=1}^{N} \ell_i c_i}{N^{-1} \sum_{i=1}^{N} c_i^2}.$$

*Proof.* See §E.3.1.

In particular, we expect this bound to become tight when the labelling budget is small relative to the pool size and the proposal is not overly concentrated on a small subset of points (see §E.3.1 for further discussion). Additionally, we empirically verify the suitability of $\lambda^\dagger$ as an alternative to $\lambda^\star$ in §C.1.

Crucially, $\lambda^\dagger$ can be estimated online during active testing. As the denominator is known on the full pool, it suffices to estimate the numerator, which we do via its LURE estimate, giving the plug-in estimate

$$\widehat{\lambda}_M = \frac{M^{-1} \sum_{m=1}^{M} V_m \ell_{I_m} c_{I_m}}{N^{-1} \sum_{i=1}^{N} c_i^2}. \tag{15}$$

While using this plug-in estimate of $\lambda$ yields a biased estimator of the risk, we show in §E.5.4 that the resulting estimator remains consistent and its bias vanishes at a rate of $O(M^{-1})$. In practice, updating $\widehat{\lambda}_M$ after every label can be unstable, since each update changes the residualised loss used for sampling; we therefore update only after every $k$ newly collected labels, keeping $\lambda$ fixed in between.

## 4 Asymptotic Properties and Approximate Confidence Intervals

While the previous sections have focused on point estimation of the risk, in practice it is typically beneficial to establish the estimator's asymptotic behavior and provide a measure of uncertainty

around this estimate. After confirming the consistency of our estimator, we therefore turn to the construction of asymptotically valid confidence intervals (CIs) for PPAT – an aspect that has been largely neglected in prior work on active testing.[6]

All expectations and probabilities for the following result are over the active sampling process, conditionally on the finite test pool. For each budget $M$, let $N_M$ denote the pool size and $R_M = N_M^{-1} \sum_{i=1}^{N_M} \ell_i$ the test pool risk. We write $\widehat{R}_{M,\mathrm{PPAT}}(\lambda)$ for the PPAT estimator at budget $M$ and tuning parameter $\lambda \in \mathbb{R}$, with $\widehat{R}_{M,\mathrm{LURE}} = \widehat{R}_{M,\mathrm{PPAT}}(0)$ the corresponding LURE estimator. We state the main result informally below; its full statement and proof are deferred to §E.

**Theorem 4.1** (Consistency and asymptotic normality of PPAT, informal)**.** *Fix $\lambda \in \mathbb{R}$. Suppose that our proposals satisfy a uniform overlap condition (i.e. there exists $\beta > 0$ s.t., for all sufficiently large $M$, $Q_m(i) \geq \beta/N_M$ for all $1 \leq m \leq M$ and $i \in S_m$) and that $N_M/M \to \alpha > 1$. Then:*

1. (Consistency.) *Assume the finite-pool averages $N_M^{-1} \sum_{i=1}^{N_M} \ell_i^2$ and $N_M^{-1} \sum_{i=1}^{N_M} \widetilde{\ell}_i^2$ are both bounded. Then we have $\widehat{R}_{M,\mathrm{PPAT}}(\lambda) - R_M \to_p 0$.*

2. (Asymptotic normality.) *Assume the finite-pool averages $N_M^{-1} \sum_{i=1}^{N_M} \ell_i^4$ and $N_M^{-1} \sum_{i=1}^{N_M} \widetilde{\ell}_i^4$ are both bounded. Write $\zeta_i(\lambda) = \ell_i - \lambda(\widetilde{\ell}_i - \widetilde{R}_M)$ for the residualised loss, with $\widetilde{R}_M = N_M^{-1} \sum_{i=1}^{N_M} \widetilde{\ell}_i$, and let $\gamma_{M,m} = \frac{N_M(N_M-M)}{(N_M-m)(N_M-m+1)}$. Assume the predictable quadratic variation*

$$\sigma_M^2(\lambda) = \frac{1}{M} \sum_{m=1}^{M} \gamma_{M,m}^2 \, s_m^2(\lambda), \qquad s_m^2(\lambda) = \frac{1}{N_M^2} \left[ \sum_{i \in S_m} \frac{\zeta_i(\lambda)^2}{Q_m(i)} - \Big( \sum_{i \in S_m} \zeta_i(\lambda) \Big)^2 \right], \quad (16)$$

*converges in probability to some $\sigma_{\mathrm{PPAT}}^2(\lambda) \in (0, \infty)$. Then*

$$\sqrt{M} \big\{ \widehat{R}_{M,\mathrm{PPAT}}(\lambda) - R_M \big\} \;\Rightarrow\; \mathcal{N}\big(0, \sigma_{\mathrm{PPAT}}^2(\lambda)\big),$$

*and $\sigma_{\mathrm{PPAT}}^2(\lambda)$ admits the plug-in estimator*

$$\widehat{\sigma}_M^2(\lambda) = \frac{1}{M} \sum_{m=1}^{M} \gamma_{M,m}^2 \big( A_m(\lambda) - \widehat{R}_{M,\mathrm{PPAT}}(\lambda) \big)^2, \qquad A_m(\lambda) = \frac{1}{N_M} \Big( \frac{\zeta_{I_m}(\lambda)}{Q_m(I_m)} + \sum_{t=1}^{m-1} \zeta_{I_t}(\lambda) \Big),$$

*which is computable from the queried labels and satisfies $\widehat{\sigma}_M^2(\lambda) \to_p \sigma_{\mathrm{PPAT}}^2(\lambda)$.*

*Moreover, under the additional conditions in §E.5.4, analogous consistency and asymptotic normality results hold when $\lambda$ is replaced by the plug-in estimate $\widehat{\lambda}_M$ from (15).*

*Proof.* See §E.5.3-E.5.4.

Crucially, Thm. 4.1 immediately yields the asymptotic $(1-\delta)$-CI

$$\widehat{R}_{M,\mathrm{PPAT}}(\lambda) \pm z_{1-\delta/2} \, \widehat{\sigma}_M(\lambda)/\sqrt{M}, \quad (17)$$

where $z_{1-\delta/2}$ denotes the $(1-\delta/2)$-quantile of the standard normal distribution. As LURE is the special case $\lambda = 0$, this also yields, as a by-product, an asymptotic CI for LURE. For a detailed discussion of the assumptions required for this result, see §E.5.5.

## 5 Experiments

We evaluate our approach on real–world regression and classification datasets. We provide a description of our experimental setup and results below with full details in §B. Code for the experiments is available on https://github.com/kiaashour/PPAT.

### 5.1 Experimental Setup

**Datasets.** For regression, we use the following UCI datasets [19]: `Keggundirected`, `Keggdirected`, `Sml`, `Bike`. For each dataset, we randomly sample a fixed training set of 250 points

[6] A notable exception is Berrada et al. [10], who bootstraped the reweighted losses to obtain confidence intervals for LURE. However, as the reweighted losses are not i.i.d. their intervals lack any theoretical guarantees.

to train the model $f$ whose risk we wish to estimate, and then randomly sample up to 5,000 of the remaining points without replacement to form the test pool. For classification, we use `CIFAR-10`, `CIFAR-100` [41], and `Tiny-ImageNet` [43]. We randomly repartition the combined original training and test sets into a training set and a test pool of the original sizes.

**Models and proxies.** For regression, $f$ is a Gaussian process regression model with an RBF kernel whose hyperparameters are optimised using the marginal likelihood, and the active-testing surrogate is a Bayesian linear regression model with the default `Scikit-Learn` hyperparameters [57]. For classification, we follow the standard pretrained-encoder pipeline: inputs are mapped to CLIP embeddings [61], and $f$ is a linear classifier trained on these frozen embeddings. The surrogate is a Laplace-approximated Bayesian neural network [17, 46]: an MLP on CLIP embeddings with three hidden layers of width 128 for `CIFAR-10` and `CIFAR-100`, and five hidden layers of width 128 for `Tiny-ImageNet`.

For our proxy, we use strong off-the-shelf predictors suited to each task. For regression, we use the `TabPFN-2.5` foundation model [26] and for classification we use the zero-shot predictions of the `ViT-L-14` CLIP model from the `sentence-transformers` library [62]. Further implementation details and ablations over surrogate and proxy choices are deferred to §B and C respectively.

**Baselines.** We compare PPAT with LURE [39], ASE [40], and Random, which samples randomly without replacement and uses the unweighted estimator in (2). Within each experimental setting, all active methods are run with the same surrogate model. For PPAT, we report the fixed choices $\lambda = 1$ and $\lambda = 0.5$, as well as the plug-in estimate $\widehat{\lambda}_M$, which we initialise at $0.5$ and update after every 100 acquired points. For ASE, we use 100 Monte Carlo samples to compute the acquisition scores; further details about the baselines are given in §A and §B.

**Active-testing setup and metrics.** In all experiments, we use a labelling budget of $M = 500$ and run 1000 independent trials. For regression, the loss is squared error and the surrogate is updated after every newly acquired label. For classification, the loss is cross-entropy and, following prior work [39, 40], we keep the surrogate fixed during acquisition. Following [39, 40], we report the median squared error, where the error is the difference between the estimated risk and the true risk on the full test pool. Mean error, confidence-interval widths, and further ablations are provided in §C.

### 5.2 PPAT Improves Risk Estimation

**Regression.** Fig. 1 shows that, across all datasets, PPAT achieves lower median squared error than Random, LURE, and ASE for all three choices of $\lambda$. The only exception occurs at very small labelling budgets, where ASE can sometimes attain lower error, for example on `Keggundirected`. However, this comes at the cost of using a **biased** risk estimate.

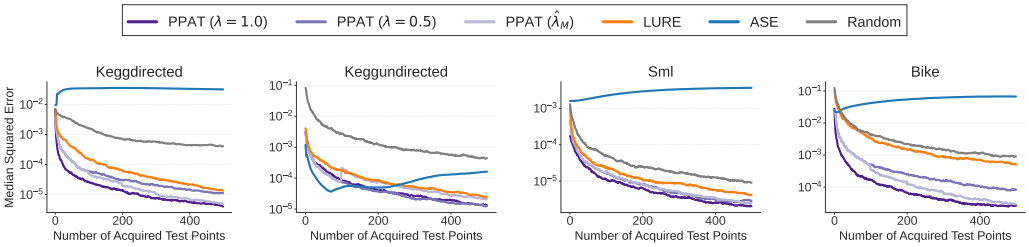



Figure 1: Regression experiments comparing PPAT, LURE, ASE, and Random. Plots show the median squared error across 1000 trials.

For the plug-in choice $\widehat{\lambda}_M$, we occasionally observe a small transient increase in error at early labelling budgets (e.g. on `Keggundirected`). This is expected, since $\widehat{\lambda}_M$ is updated from the actively collected labels during acquisition, which introduces finite-sample bias relative to fixed-$\lambda$ PPAT and also changes the proposal used to sample future points. As the number of acquired labels grows, however, $\widehat{\lambda}_M$ stabilises and this bias diminishes (see §C.3.1). Importantly, we also see that PPAT with $\widehat{\lambda}_M$ generally converges to PPAT with the best performing fixed $\lambda$, suggesting that targeting $\lambda^\dagger$ is an effective choice (see also §C.1).

**Classification.** Fig. 2 shows the same pattern on the classification datasets: PPAT consistently achieves lower median squared error than Random, LURE, and ASE for all choices of $\lambda$, with the exception of `Tiny-ImageNet`, where PPAT with $\lambda = 1$ performs comparably to Random. We further note that Random outperforms LURE on these datasets, likely a result of the surrogate poorly tracking the losses. Indeed, since LURE builds its proposal, and hence its importance weights, from the surrogate's expected losses, a surrogate that poorly tracks the true losses can result in importance weights that inflate the variance of LURE. In contrast, PPAT is less affected by this issue: it importance-weights the residualised loss rather than the raw loss, and since the proxy explains part of the loss variation, the residuals have smaller spread, making the variance less sensitive to errors in the surrogate.

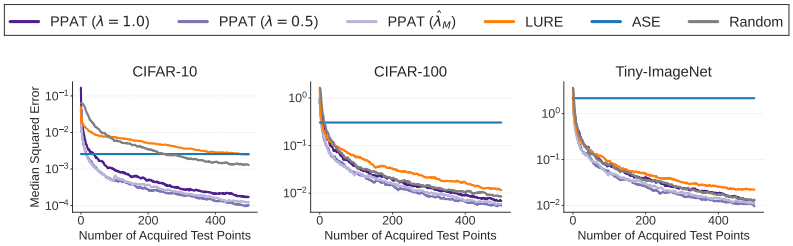



Figure 2: Classification experiments comparing PPAT, LURE, ASE, and Random. Plots show the median squared error across 1000 trials.

Overall, these results indicate that PPAT improves risk estimation not only for tabular regression, but also for more challenging, higher-dimensional image-classification tasks.

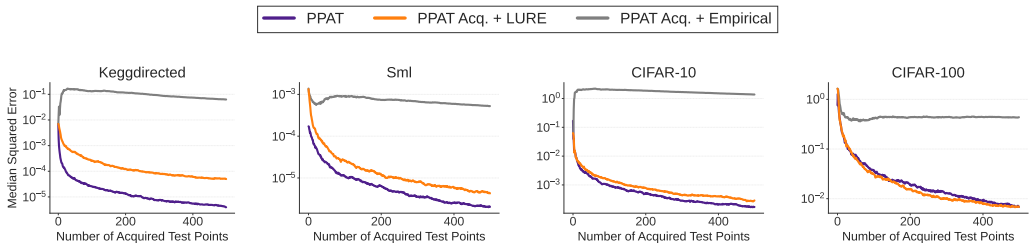



Figure 3: Experiments on UCI and classification datasets studying the influence of our estimator. We compare PPAT with $\lambda = 1$ to using the PPAT acquisition strategy, $Q^{\text{PPAT}}$, but with the LURE estimate of the risk (PPAT Acq. + LURE) and the empirical estimate of the risk (PPAT Acq. + Empirical). Plots show the median squared error across 1000 trials.

### 5.3 Disentangling the Components of PPAT

Beyond the choice of $\lambda$, which we have discussed in §5.2 and §C.1, there are two main components that can affect the behaviour of PPAT: the choice of acquisition strategy and the choice of the PPI-like estimator. To disentangle their contributions, we study the effect of using different estimators in Fig. 3 and different acquisition strategies in Fig. 4 on four representative datasets.

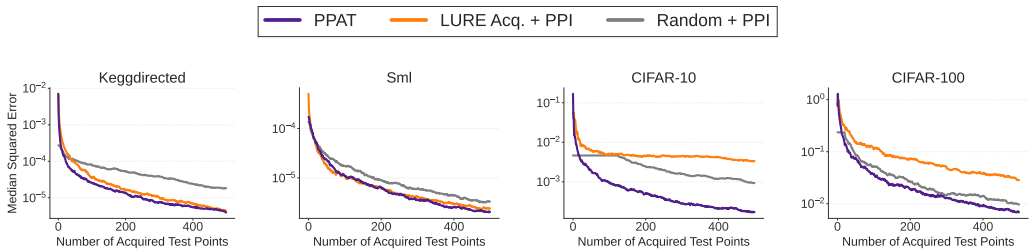



Figure 4: Experiments on UCI and classification datasets studying the influence of our acquisition strategy. We compare PPAT with $\lambda = 1$ to using the PPI estimator with the LURE acquisition strategy (LURE Acq. + PPI) and random sampling (Random + PPI). Plots show the median squared error across 1000 trials.

The notable improvements observed over both different acquisition strategies (Fig. 4) and different estimators (Fig. 3) show that the gains of PPAT do not come merely from using a better, PPI-like estimator. Instead, they come from combining our PPI-like estimator with an acquisition strategy tailored to the corresponding residualised objective.

### 5.4 Coverage

Beyond reducing estimation error, we also want our CIs to be well-calibrated: across repeated active-testing runs, a nominal $(1-\delta)$-CI should contain the true test-pool risk with frequency close to $(1-\delta)$. Fig. 5 shows the coverage for the CIs presented in §4 for PPAT, LURE, and Random on four representative datasets. ASE is excluded as it **cannot** provide asymptotically valid CIs without additional assumptions about $f$ and $\pi_m(\cdot|\mathbf{x})$. We see that on all datasets, our approach achieves the desired coverage level for all choices of $\lambda$. More importantly, it achieves the desired coverage at either the same or a significantly faster rate than all other approaches. Additionally, our approach attains significantly smaller widths for the confidence intervals than the competing approaches (see §C.3.2).

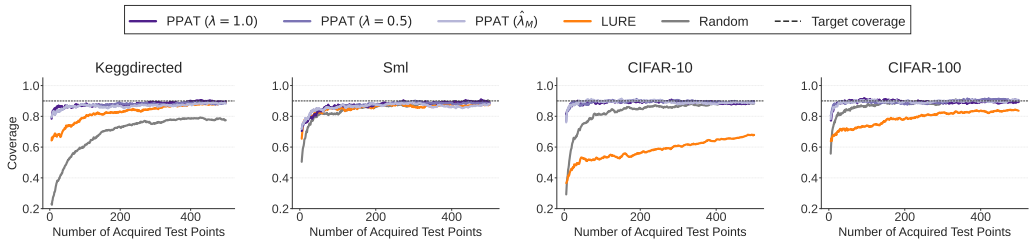



Figure 5: Representative experiments on UCI and classification datasets for the coverage of PPAT, Random, LURE. Plots show the coverage of asymptotic confidence intervals at a target error level of $\delta = 0.1$ across 1000 trials.

## 6 Related Work

We review the most closely related work here, and defer a more extensive discussion to §A.

PPAT builds directly on the active–testing line of work initiated by [22], who introduced the *Levelled Unbiased Risk Estimator (LURE)*, and [39], who developed it into a sample–efficient model–evaluation framework. A closely related approach is that of *Active Surrogate Estimators (ASE)* [40], which, like PPAT, exploit predictions over the entire pool. ASE, however, does so in a fundamentally different way: rather than correcting an importance–weighted estimator with observed labels, it learns a surrogate for the conditional loss and imputes losses across the pool via interpolation. While this can be successful at very low labelling budgets, ASE loses the theoretical guarantees of LURE, since surrogate misspecification or finite–sample estimation error enters the risk estimate directly. More recently, [10] scaled LURE–based active testing to large language model (LLM) evaluation using fixed, in–context–learning surrogates. Our approach is complementary: it can incorporate the same in–context surrogates while additionally leveraging cheap black–box predictions to improve the estimator itself.

## 7 Discussion

In this work, we introduced **Prediction-Powered Active Testing (PPAT)**, a label-efficient framework for risk estimation that combines the unbiased LURE estimator with a prediction-powered control variate. Rather than imputing labels, PPAT uses cheap proxy predictions to residualise the loss, preserving unbiasedness while reducing variance. We also established asymptotic normality for PPAT, yielding asymptotic confidence intervals that provide a principled quantification of uncertainty around the risk estimate. Our results show that PPAT improves risk estimation across tabular regression and image-classification tasks, while attaining target coverage with substantially fewer labels and producing narrower confidence intervals than competing approaches. We defer a discussion of the limitations of our approach and directions for future work to §D.4.

## Acknowledgments

KA is supported by the EPSRC Centre for Doctoral Training in Modern Statistics and Statistical Machine Learning (EP/S023151/1). VK is supported by the Clarendon Funds Scholarship. DH is supported by the Research Council of Finland (Flagship programme: Finnish Center for Artificial Intelligence FCAI, 359207). TR is supported by the EPSRC grant EP/Y037200/1. The authors are grateful to Stefano Cortinovis for helpful discussions.

## References

[1] Josh Abramson, Jonas Adler, Jack Dunger, Richard Evans, Tim Green, Alexander Pritzel, Olaf Ronneberger, Lindsay Willmore, Andrew J Ballard, Joshua Bambrick, Sebastian Bodenstein, David A Evans, Chia-Chun Hung, Michael O'Neill, David Reiman, Kathryn Tunyasuvunakool, Zachary Wu, Akvile Zemgulyte, Eirini Arvaniti, Charles Beattie, Ottavia Bertolli, Alex Bridgland, Alexey Cherepanov, Miles Congreve, Alexander Imani Cowen-Rivers, Andrew Cowie, Michael Figurnov, Fabian B Fuchs, Hannah Gladman, Rishub Jain, Yousuf A. Khan, Caroline M R Low, Kuba Perlin, Anna Potapenko, Pascal Savy, Sukhdeep Singh, Adrian Stecula, Ashok Thillaisundaram, Catherine Tong, Sergei Yakneen, Ellen D. Zhong, Michal Zielinski, Augustin Žídek, Vic-613 tor Bapst, Pushmeet Kohli, Max Jaderberg, Demis Hassabis, and John M. Jumper. Accurate structure prediction of biomolecular interactions with alphafold 3. *Nature*, 630:493 – 500, 2024. URL https://api.semanticscholar.org/CorpusID:269633210. 1

[2] Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. *arXiv preprint arXiv:1610.01644*, 2016. 21

[3] Anastasios N Angelopoulos, Stephen Bates, Clara Fannjiang, Michael I Jordan, and Tijana Zrnic. Prediction-powered inference. *Science*, 382(6671):669–674, 2023. 1, 3, 5, 18, 22

[4] Anastasios N Angelopoulos, John C Duchi, and Tijana Zrnic. Ppi++: Efficient prediction-powered inference. *arXiv preprint arXiv:2311.01453*, 2023. 3, 18

[5] Dana Angluin. Queries and concept learning. *Mach. Learn.*, 2(4):319–342, April 1988. ISSN 0885-6125. doi: 10.1023/A:1022821128753. URL https://doi.org/10.1023/A:1022821128753. 1

[6] Shir Ashury-Tahan, Ariel Gera, Benjamin Sznajder, Leshem Choshen, Liat Ein-Dor, and Eyal Shnarch. Label-efficient model selection for text generation. *arXiv preprint arXiv:2402.07891*, 2024. 18

[7] Les Atlas, David Cohn, and Richard Ladner. Training connectionist networks with queries and selective sampling. In D. Touretzky, editor, *Advances in Neural Information Processing Systems*, volume 2. Morgan-Kaufmann, 1989. URL https://proceedings.neurips.cc/paper_files/paper/1989/file/b1a59b315fc9a3002ce38bbe070ec3f5-Paper.pdf. 1

[8] Heejung Bang and James M. Robins. Doubly robust estimation in missing data and causal inference models. *Biometrics*, 61(4):962–973, 12 2005. ISSN 0006-341X. doi: 10.1111/j.1541-0420.2005.00377.x. URL https://doi.org/10.1111/j.1541-0420.2005.00377.x. 19

[9] Paul N. Bennett and Vitor R. Carvalho. Online stratified sampling: evaluating classifiers at web-scale. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, CIKM '10, page 1581–1584, New York, NY, USA, 2010. Association for Computing Machinery. ISBN 9781450300995. doi: 10.1145/1871437.1871677. URL https://doi.org/10.1145/1871437.1871677. 18

[10] Gabrielle Berrada, Jannik Kossen, Freddie Bickford Smith, Muhammed Razzak, Yarin Gal, and Tom Rainforth. Scaling up active testing to large language models. *arXiv preprint arXiv:2508.09093*, 2025. 1, 7, 10

[11] Lukas Biewald. Experiment tracking with weights and biases, 2020. URL https://www.wandb.com/. Software available from wandb.com. 22

[12] Patrick Billingsley. *Probability and Measure*. John Wiley & Sons, New York, 3 edition, 1995. 46

[13] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 1

[14] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PmLR, 2020. 21

[15] William G. Cochran. *Sampling Techniques*. John Wiley & Sons, 3 edition, 1977. 19

[16] Gabriele Corso, Hannes Stärk, Bowen Jing, Regina Barzilay, and Tommi Jaakkola. Diffdock: Diffusion steps, twists, and turns for molecular docking. *arXiv preprint arXiv:2210.01776*, 2022. 1

[17] Erik Daxberger, Agustinus Kristiadi, Alexander Immer, Runa Eschenhagen, Matthias Bauer, and Philipp Hennig. Laplace redux-effortless bayesian deep learning. *Advances in neural information processing systems*, 34:20089–20103, 2021. 8, 21, 22

[18] Shrey Desai and Greg Durrett. Calibration of pre-trained transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 295–302, 2020. 34

[19] Dua Dheeru and Efi Karra Taniskidou. UCI machine learning repository, 2017. URL `http://archive.ics.uci.edu/ml`. 7

[20] Trefor Evans. uci_datasets: Regression datasets from the uci repository with standardized test-train splits. `https://github.com/treforevans/uci_datasets`, 2022. 22

[21] Hadi Fanaee-T. Bike Sharing. UCI Machine Learning Repository, 2013. DOI: https://doi.org/10.24432/C5W894. 23

[22] Sebastian Farquhar, Yarin Gal, and Tom Rainforth. On statistical bias in active learning: How and when to fix it. *arXiv preprint arXiv:2101.11665*, 2021. 1, 2, 3, 10, 35, 57

[23] Adam Fisch, Joshua Maynez, R Hofer, Bhuwan Dhingra, Amir Globerson, and William W Cohen. Stratified prediction-powered inference for effective hybrid evaluation of language models. *Advances in Neural Information Processing Systems*, 37:111489–111514, 2024. 19

[24] Paul Glasserman. *Monte Carlo Methods in Financial Engineering*, volume 53. Springer, New York, 2004. 3

[25] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems*, 33:21271–21284, 2020. 21

[26] Léo Grinsztajn, Klemens Flöge, Oscar Key, Felix Birkel, Philipp Jund, Brendan Roof, Benjamin Jäger, Dominik Safaric, Simone Alessi, Adrian Hayler, et al. Tabpfn-2.5: Advancing the state of the art in tabular foundation models. *arXiv preprint arXiv:2511.08667*, 2025. 8, 20

[27] Peter Hall, Eugene Lukács, Zygmunt William Birnbaum, and Chris C. Heyde. Martingale limit theory and its application. 1980. URL `https://api.semanticscholar.org/CorpusID:120148048`. 57

[28] Max Hamilton, Jinlin Lai, Wenlong Zhao, Subhransu Maji, and Daniel Sheldon. Active measurement: Efficient estimation at scale. *arXiv preprint arXiv:2507.01372*, 2025. 18

[29] Charles R Harris, K Jarrod Millman, Stéfan J Van Der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J Smith, et al. Array programming with numpy. *nature*, 585(7825):357–362, 2020. 22

[30] Yuheng Huang, Jiayang Song, Qiang Hu, Felix Juefei-Xu, and Lei Ma. Actracer: Active testing of large language model via multi-stage sampling. *ACM Transactions on Software Engineering and Methodology*, 35(3):1–30, 2026. 18

[31] J. D. Hunter. Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, 9 (3):90–95, 2007. doi: 10.1109/MCSE.2007.55. 22

[32] Alexander Immer, Maciej Korzepa, and Matthias Bauer. Improving predictions of bayesian neural nets via local linearization. In *International conference on artificial intelligence and statistics*, pages 703–711. PMLR, 2021. 21

[33] Disi Ji, Robert L Logan, Padhraic Smyth, and Mark Steyvers. Active bayesian assessment of black-box classifiers. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 7935–7944, 2021. 18

[34] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*, pages 4904–4916. PMLR, 2021. 1

[35] John M. Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, Alex Bridgland, Clemens Meyer, Simon A A Kohl, Andy Ballard, Andrew Cowie, Bernardino Romera-Paredes, Stanislav Nikolov, Rishub Jain, Jonas Adler, Trevor Back, Stig Petersen, David Reiman, Ellen Clancy, Michal Zielinski, Martin Steinegger, Michalina Pacholska, Tamas Berghammer, Sebastian Bodenstein, David Silver, Oriol Vinyals, Andrew W. Senior, Koray Kavukcuoglu, Pushmeet Kohli, and Demis Hassabis. Highly accurate protein structure prediction with alphafold. *Nature*, 596:583 – 589, 2021. URL https://api.semanticscholar.org/CorpusID:235959867. 1

[36] Namit Katariya, Arun Iyer, and Sunita Sarawagi. Active evaluation of classifiers on large datasets. In *2012 IEEE 12th International Conference on Data Mining*, pages 329–338, 2012. doi: 10.1109/ICDM.2012.161. 1, 18

[37] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 21

[38] Simon Kornblith, Jonathon Shlens, and Quoc V Le. Do better imagenet models transfer better? In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2661–2671, 2019. 21

[39] Jannik Kossen, Sebastian Farquhar, Yarin Gal, and Tom Rainforth. Active testing: Sample-efficient model evaluation. In *International Conference on Machine Learning*, pages 5753–5763. PMLR, 2021. 1, 2, 3, 5, 8, 10, 18, 20, 22, 29, 32

[40] Jannik Kossen, Sebastian Farquhar, Yarin Gal, and Thomas Rainforth. Active surrogate estimators: An active learning approach to label-efficient model evaluation. *Advances in Neural Information Processing Systems*, 35:24557–24570, 2022. 1, 3, 8, 10, 20, 22, 29, 33

[41] Alex Krizhevsky. Learning multiple layers of features from tiny images. pages 32–33, 2009. URL https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf. 8, 21, 23

[42] Anurag Kumar and Bhiksha Raj. Classifier risk estimation under limited labeling resources. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 3–15. Springer, 2018. 18

[43] Ya Le and Xuan S. Yang. Tiny imagenet visual recognition challenge. 2015. URL https://api.semanticscholar.org/CorpusID:16664790. 8, 21, 23

[44] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR, 2023. 1

[45] Peng Liu, Lizhe Wang, Rajiv Ranjan, Guojin He, and Lei Zhao. A survey on active deep learning: From model driven to data driven. *ACM Comput. Surv.*, 54(10s), September 2022. ISSN 0360-0300. doi: 10.1145/3510414. URL https://doi.org/10.1145/3510414. 1

[46] David J. C. MacKay. Bayesian interpolation. *Neural Computation*, 4(3):415–447, 05 1992. ISSN 0899-7667. doi: 10.1162/neco.1992.4.3.415. URL https://doi.org/10.1162/neco.1992.4.3.415. 8, 21

[47] TorchVision maintainers and contributors. Torchvision: Pytorch's computer vision library. https://github.com/pytorch/vision, 2016. 21, 22

[48] Mitsuru Matsuura and Satoshi Hara. Active model selection: A variance minimization approach. In *NeurIPS 2023 Workshop on Adaptive Experimental Design and Active Learning in the Real World*, 2023. URL https://openreview.net/forum?id=vBwfTUDTtz. 18

[49] Joshua Maynez, Priyanka Agrawal, and Sebastian Gehrmann. Benchmarking large language model capabilities for conditional generation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9194–9213, 2023. 18

[50] Wes McKinney. Data structures for statistical computing in python. *SciPy 2010*, 2010. doi: 10.25080/Majora-92bf1922-00a. URL https://doi.org/10.25080/Majora-92bf1922-00a. 22

[51] Matthias Minderer, Josip Djolonga, Rob Romijnders, Frances Hubis, Xiaohua Zhai, Neil Houlsby, Dustin Tran, and Mario Lucic. Revisiting the calibration of modern neural networks. *Advances in neural information processing systems*, 34:15682–15694, 2021. 34

[52] Muhammad Naeem and Sohail Asghar. KEGG Metabolic Relation Network (Directed). UCI Machine Learning Repository, 2011. DOI: https://doi.org/10.24432/C5CK52. 23

[53] Muhammad Naeem and Sohail Asghar. KEGG Metabolic Reaction Network (Undirected). UCI Machine Learning Repository, 2011. DOI: https://doi.org/10.24432/C5G609. 23

[54] Phuc Nguyen, Deva Ramanan, and Charless Fowlkes. Active testing: An efficient and robust framework for estimating accuracy. In *International Conference on Machine Learning*, pages 3759–3768. PMLR, 2018. 18

[55] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 21

[56] The pandas development team. pandas-dev/pandas: Pandas, February 2020. URL https://doi.org/10.5281/zenodo.3509134. 22

[57] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011. 8, 20, 21, 22, 24

[58] Jo Plested, Musa Phiri, and Tom Gedeon. Deep transfer learning for image classification: a survey. *Artificial Intelligence Review*, 2026. 21

[59] Felipe Maia Polo, Lucas Weber, Leshem Choshen, Yuekai Sun, Gongjun Xu, and Mikhail Yurochkin. tinybenchmarks: evaluating llms with fewer examples. *arXiv preprint arXiv:2402.14992*, 2024. 18

[60] Alvaro Prat, Leo Zhang, Charlotte M Deane, Yee Whye Teh, and Garrett M Morris. Sigmadock: Untwisting molecular docking with fragment-based se (3) diffusion. *arXiv preprint arXiv:2511.04854*, 2025. 1

[61] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763, 2021. URL https://proceedings.mlr.press/v139/radford21a.html. 1, 8, 21, 22

[62] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 11 2019. URL https://arxiv.org/abs/1908.10084. 8, 21, 22, 24

[63] Hippolyt Ritter, Aleksandar Botev, and David Barber. A scalable laplace approximation for neural networks. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=Skdvd2xAZ. 21

[64] James M. Robins, Andrea Rotnitzky, and Lue Ping Zhao. Estimation of regression coefficients when some regressors are not always observed. *Journal of the American Statistical Association*, 89(427):846–866, 1994. ISSN 01621459, 1537274X. URL http://www.jstor.org/stable/2290910. 19

[65] Pablo Romeu-Guallart and Francisco Zamora-Martinez. SML2010. UCI Machine Learning Repository, 2014. DOI: https://doi.org/10.24432/C5RS3S. 23

[66] Gayathri Saranathan, Mahammad Parwez Alam, James Lim, Suparna Bhattacharya, Soon Yee Wong, Martin Foltin, and Cong Xu. DELE: Data efficient LLM evaluation. In *ICLR 2024 Workshop on Navigating and Addressing Data Problems for Foundation Models*, 2024. URL https://openreview.net/forum?id=I8bsxPWLNF. 18

[67] Christoph Sawade, Niels Landwehr, Steffen Bickel, and Tobias Scheffer. Active risk estimation. In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, ICML'10, page 951–958, Madison, WI, USA, 2010. Omnipress. ISBN 9781605589077. 1, 18

[68] Christoph Sawade, Niels Landwehr, and Tobias Scheffer. Active estimation of f-measures. In J. Lafferty, C. Williams, J. Shawe-Taylor, R. Zemel, and A. Culotta, editors, *Advances in Neural Information Processing Systems*, volume 23. Curran Associates, Inc., 2010. URL https://proceedings.neurips.cc/paper_files/paper/2010/file/d7a728a67d909e714c0774e22cb806f2-Paper.pdf. 1

[69] Burr Settles. Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison, 2009. URL http://axon.cs.byu.edu/~martinez/classes/778/Papers/settles.activelearning.pdf. 1

[70] Maria-Eleni Sfyraki and Jun-Kun Wang. Revisiting active sequential prediction-powered mean estimation. *arXiv preprint arXiv:2604.18569*, 2026. 18

[71] Sanbao Su, Xin Li, Thang Doan, Sima Behpour, Wenbin He, Liang Gou, Fei Miao, and Liu Ren. Metaat: Active testing for label-efficient evaluation of dense recognition tasks. In *European Conference on Computer Vision*, pages 325–342. Springer, 2024. 18

[72] Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023. 1

[73] Michael E. Tipping. Sparse bayesian learning and the relevance vector machine. *J. Mach. Learn. Res.*, 1:211–244, September 2001. ISSN 1532-4435. doi: 10.1162/15324430152748236. URL https://doi.org/10.1162/15324430152748236. 20

[74] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023. 1

[75] Dustin Tran, Jeremiah Liu, Michael W Dusenberry, Du Phan, Mark Collier, Jie Ren, Kehang Han, Zi Wang, Zelda Mariet, Huiyi Hu, et al. Plex: Towards reliability using pretrained large model extensions. *arXiv preprint arXiv:2207.07411*, 2022. 34

[76] Pauli Virtanen, Ralf Gommers, Travis E Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, et al. Scipy 1.0: fundamental algorithms for scientific computing in python. *Nature methods*, 17(3):261–272, 2020. 22

[77] Rajan Vivek, Kawin Ethayarajh, Diyi Yang, and Douwe Kiela. Anchor points: Benchmarking models with much fewer examples. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1576–1601, 2024. 18

[78] Shuohang Wang, Yang Liu, Yichong Xu, Chenguang Zhu, and Michael Zeng. Want to reduce labeling cost? gpt-3 can help. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4195–4205, 2021. 1

[79] Ross Wightman. PyTorch Image Models. `https://github.com/huggingface/pytorch-image-models`, 2019. 22

[80] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.emnlp-demos.6. URL `https://aclanthology.org/2020.emnlp-demos.6`. 22

[81] Emine Yilmaz, Peter Hayes, Raza Habib, Jordan Burgess, and David Barber. Sample efficient model evaluation. *arXiv preprint arXiv:2109.12043*, 2021. 1, 18

[82] Dayou Yu, Weishi Shi, and Qi Yu. Actively testing your model while it learns: Realizing label-efficient learning in practice. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 31404–31427. Curran Associates, Inc., 2023. URL `https://proceedings.neurips.cc/paper_files/paper/2023/file/63ef323523f3be8b58ed9277cc747485-Paper-Conference.pdf`. 1, 18

[83] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11975–11986, 2023. 1

[84] Tijana Zrnic and Emmanuel J Candès. Active statistical inference. *arXiv preprint arXiv:2403.03208*, 2024. 18, 29

[85] Tijana Zrnic and Emmanuel J Candès. Cross-prediction-powered inference. *Proceedings of the National Academy of Sciences*, 121(15):e2322083121, 2024. 19

## Appendix Contents


**A Extended Related Work** **18**

**B Further Details on Experimental Setup** **20**

B.1 Section 5.2 . . . . . 20

B.2 Section 5.3 . . . . . 22

B.3 Section 5.4 . . . . . 22

B.4 Computational Resources and Licenses . . . . . 22

**C Additional Results** **24**

C.1 Empirical Verification of $\lambda^{\dagger}$ . . . . . 24

C.2 Additional Ablations . . . . . 24

C.3 Additional Metrics . . . . . 26

C.4 Additional Comparisons . . . . . 27

C.5 On Variance Reduction over LURE . . . . . 29

**D Additional Details and Discussions** **32**

D.1 On the Choice of the Surrogate Score in Section 2.1 . . . . . 32

D.2 Simplified Expressions for the Proposals of LURE and PPAT . . . . . 32

D.3 Extended Background . . . . . 33

D.4 Future Work and Limitations . . . . . 34

**E Mathematical Results** **35**

E.1 Definitions . . . . . 35

E.2 Generalising LURE with Triangular–Arrays . . . . . 35

E.3 Finite–Sample Unbiasedness and Variance . . . . . 37

E.4 Myopic Oracle Proposals . . . . . 43

E.5 Asymptotics . . . . . 44

# A Extended Related Work

Here, we provide a more detailed discussion of related work.

**Earlier work on label–efficient evaluation.** Earlier work on label–efficient model evaluation considered more restricted sampling schemes. Stratification–based methods [9, 33, 36, 42] partition the test pool into strata, for example using a simple non–adaptive measure of model confidence, and sample uniformly within each. PPAT is complementary here: once a stratification is fixed, it can be applied within each stratum to further improve label efficiency. A different line of work considered importance and Poisson sampling: [67] used importance sampling *with* replacement, which is suboptimal relative to estimators that explicitly account for sampling without replacement (see Appendix D of [39]); [81] proposed a Poisson–sampling approach, but it is restricted to particular evaluation metrics, uses a non–adaptive acquisition rule, and relies on ratio–type estimates that are generally biased. [54] studied a more specialised active–testing setting focused on human vetting of noisy labels.

**Applications and extensions of active testing.** Active testing has recently been extended to a range of settings. In the LLM evaluation setting, [30] proposed *AcTracer* which partitions the test pool via hidden–state representations of the target LLM and performs multi–stage sampling using online variance estimates and confidence traces. [71] addressed dense computer vision tasks such as segmentation and object detection. [82] incorporated active testing into model training, and [6, 48] studied active testing for model selection.

**Benchmark subsampling.** A related but distinct line of work reduces evaluation cost by constructing smaller test sets from existing *labelled* LLM benchmarks. [49] showed that uniform subsampling can preserve stable model rankings in some settings, and [59, 66, 77] developed more sophisticated dataset–reduction methods. These approaches are complementary to ours in their goal of reducing evaluation cost, but address a different setting: they assume an already labelled test set and aim to support comparisons across many models, whereas we study the problem of acquiring new labels to estimate the risk of a given model on a fixed unlabelled test pool.

**Active measurement.** Also related to our work is *Active Measurement* (AM, [28]), which also builds on LURE and combines model predictions with importance sampling without replacement, but targets scientific measurements (e.g., total bird counts) rather than a model's test risk. Our framework uses predictions in fundamentally different ways: AM uses AI predictions to define the proposal itself and iteratively refines the predictor with acquired labels, whereas PPAT uses a fixed proxy as a *control variate* that residualises the loss, with the proposal constructed separately from a surrogate tailored to this residualised objective. Moreover, we note that AM assumes their proxy model can be updated as more labels are acquired, which is less general than our setup[7]. In principle, AM's iterative proposal refinement and weighting schemes could also be combined with PPAT's control–variate estimator.

**Active statistical inference.** *Active statistical inference* (ASI, [84]) considers adaptive label sampling for statistical inference, proposing batch and sequential methods that yield valid confidence intervals for $M$–estimation problems. Recent work revisited the sequential mean-estimation setting and provided non–asymptotic guarantees for online policies that choose the probability of querying each ground–truth label [70]. While risk estimation is a particular instance of $M$–estimation, their framework is not framed around model evaluation, and their sequential formulation is closer to online selective labelling than to pool–based acquisition: labels are acquired point–by–point from a stream rather than selected from the full remaining pool. This differs from our setting, where we focus on pool–based risk estimation and use black–box predictions as control variates within an unbiased active–testing estimator. We compare with ASI in §C.4.3 and find that we can outperform their approach in their "active" setting while achieving similar performance in the "batch" setting.

**Prediction–powered inference and classical augmented estimators.** *Prediction–powered inference* (PPI, [3]) was introduced to construct tighter confidence intervals in the semi–supervised setting by leveraging machine learning predictions, and has since been extended in several directions: PPI++ [4] provides a computationally efficient, loss–based formulation with a power–tuning parameter $\lambda$;

[7] Indeed, our approach and theoretical results naturally transfer to the setting where our proxy model can also be updated as more labels are acquired.

Stratified PPI [23] improves efficiency via data stratification; and Cross–PPI [85] trains the predictor within the same prediction–powered pipeline. PPI is also related to classical augmented estimators: in survey sampling, regression estimators use auxiliary variables to improve the efficiency of the estimator [15]; in causal inference, augmented inverse probability weighted estimators combine inverse–probability weighting with an outcome–model correction and are doubly robust when either the propensity model or the outcome model is correctly specified [8, 64].

Our work is related to these works in spirit but differs in both objective and setting: rather than semi–supervised inference with i.i.d. labelled and unlabelled data, we study active risk estimation over a finite test pool, using black–box predictions as a control variate inside an importance–weighted active–testing estimator.

# B Further Details on Experimental Setup

Here, we provide further details for the experiments in §5.

## B.1 Section 5.2

Here, we provide details for the experiments in §5.2.

### B.1.1 Regression Experiments

**Datasets.** Below, we provide further details on the UCI datasets used. We use $n$ to denote the full dataset size and $d$ to denote the number of features.

- `Keggundirected` ($n = 63{,}608$, $d = 27$): A biology dataset derived from KEGG metabolic pathways represented as undirected reaction networks. The inputs are graph-level descriptors of pathway topology.
- `Keggdirected` ($n = 48{,}827$, $d = 20$): A related KEGG dataset where metabolic pathways are represented as directed relation networks. The features summarise structural properties of these directed pathway graphs.
- `Bike` ($n = 17{,}379$, $d = 17$): Hourly bike–rental data from the Capital Bikeshare system in 2011–2012. The covariates include weather, seasonal, and calendar information.
- `Sml` ($n = 4{,}137$, $d = 26$): A smart–home monitoring dataset collected over approximately 40 days in a domotic house. The inputs include indoor, outdoor, weather, humidity, lighting, and actuator measurements.

We load each of the above datasets from the `uci_datasets` repository. For each dataset, we randomly sample a fixed training set of 250 points which we use to train the model that we wish to evaluate. For computational efficiency, we then randomly sample 5,000 points without replacement from the remaining data to form the test pool, using all remaining points when fewer than 5,000 are available.

**Models.** We choose $f$ to be a Gaussian process regression model with a Radial Basis Function (RBF) kernel. We optimise its hyperparameters using the marginal likelihood, with the lengthscale initialised at 1.0. We use the `Scikit-Learn` implementation [57].

For our surrogate model, we use a Bayesian linear regression model [73] with the default hyperparameters from `Scikit-Learn` [57] and ablate with different choices of surrogate models in §C.2.2. We again use the `Scikit-Learn` implementation.

For the choice of $g$ for PPAT, we use the `TabPFN-2.5` [26] foundation model and ablate with different choices of $g$ in §C.2.1. To make predictions on our test pool, we use in–context learning [26], where we use our training data as the in–context data.

**Baselines.** We compare PPAT with LURE [39], ASE [40], and Random (sampling points randomly without replacement and using the unweighted estimator in (2)). All approaches use the same surrogate model. For PPAT, we compare with the three different choices of $\lambda$ described in §3.3. To compute the acquisition scores for ASE, we use 100 Monte Carlo samples for each point: 10 samples from the posterior on the model parameters, and 10 samples from the model likelihood. See §D.3.1 for more details about the ASE acquisition function.

For PPAT and LURE, we compute the acquisition scores using the simplified formulas discussed in §D.2, which can be readily computed using the mean prediction and predictive variance output by the surrogate model. For the PPAT plug–in estimate of $\lambda$, $\widehat{\lambda}_M$, we initialise it at 0.5 and update it after every 100 acquired points using (15).

**Active testing setup and metrics.** We take our loss function to be the squared error . We use a budget of $M = 500$ points and update our surrogate model after each newly acquired point. Each experiment is run for 1000 trials. Following [39, 40], we report the median squared error. The mean error is presented in §C.3.1. Here, the error is defined as the difference between our estimate of the risk and the true risk on the test pool.

### B.1.2 Classification Experiments

**Datasets.** Below, we provide further details about the classification dataset used. We provide the size of the train and test sets used as well as the resolution of the original images.

- `CIFAR-10` ([41], 50,000 train, 10,000 test; $32 \times 32 \times 3$): A natural–image classification dataset with 10 object classes. Each class contains 6,000 images.
- `CIFAR-100` ([41], 50,000 train, 10,000 test; $32 \times 32 \times 3$): A more fine–grained version of `CIFAR-10` with 100 object classes. Each class contains 600 images, making the classification task harder.
- `Tiny-ImageNet` ([43], 100,000 train, 10,000 test; $64 \times 64 \times 3$): A reduced version of ImageNet with 200 classes. It contains higher–resolution images and substantially more classes than `CIFAR-10` and `CIFAR-100`.

We load each dataset from the `torchvision` [47] library. We first combine the original training and test sets, using the validation set in place of the test set for `Tiny-ImageNet` because its original test set is unlabelled. We then randomly sample a training set without replacement from the combined data to train $f$, and randomly sample the test pool without replacement from the remaining points[8]. We retain the original training and test set sizes; for `Tiny-ImageNet`, the test pool size matches the original validation set size.

**Models.** In line with standard practice in modern image–classification pipelines [14, 25, 55, 58, 61], we use a fixed pretrained image encoder to map each input $\mathbf{x}$ to an embedding $e(\mathbf{x})$, and train lightweight prediction models on top of these embeddings. Here, the encoder $e$ is a map $e : \mathbb{R}^{d_1} \to \mathbb{R}^{d_2}$, where $d_1$ is the size of the input space, $d_2$ is the size of the embeddings and $d_2 \ll d_1$. In all of our classification experiments, we use embeddings from a CLIP image encoder [61]. Specifically, we use the `ViT-B-32` encoder with model card `openai/clip-vit-base-patch32` from the `CLIP` library.

The model $f$ whose risk we evaluate is a linear classifier trained on the CLIP embeddings. This corresponds to the standard linear probing setup, where a linear prediction head is trained on top of a frozen representation to evaluate the usefulness of the learned features [2, 14, 38]. For this, we use the logistic regression model class from `Scikit-Learn` [57], where we set the maximum number of iterations to 300 and keep the remaining hyperparameters at their default values.

For the active–testing surrogate, it is important to use a model with well–calibrated predictive uncertainties, since our acquisition rules depend on expectations under the surrogate predictive distribution $\pi_m(\cdot \mid \mathbf{x})$. We therefore use a Laplace–approximated Bayesian neural network [17, 46], implemented using the `laplace-redux` library. The surrogate is an MLP trained on the CLIP embeddings, with three hidden layers of width 128 for `CIFAR-10` and `CIFAR-100`, and five hidden layers of width 128 for `Tiny-ImageNet`. We train the network for 200 epochs using Adam [37] with batch size 64, learning rate $10^{-3}$, and weight decay $10^{-4}$. We then fit a last-layer Laplace approximation with a Kronecker–factored structure [63], optimising the prior precision by maximising the marginal likelihood. For prediction, we use the linearised predictive distribution induced by the Laplace posterior, which gives an approximate Gaussian distribution over the latent network outputs around the trained weights [32]. We then obtain class probabilities by Monte Carlo through the softmax link function using 100 samples. We again ablate with different surrogates in §C.2.2.

For the choice of $g$, we use the zero–shot predictions of a pretrained CLIP [61] model, specifically the `ViT-L-14` variant from the `sentence-transformers` library [62] with model card `ViT-L-14::datacomp_xl_s13b_b90k`. We again ablate with different choices of $g$ in §C.2.1.

**Baselines.** All our baselines use the same surrogate model and follow the same setup as for the regression datasets. To compute the acquisition scores for ASE, we use 100 samples from the posterior on the model parameters. Note that for classification problems we do not need to approximate the expectation over $Y$ as $y$ is discrete.

[8] We use random repartitioning for consistency with the regression experiments, where training sets and test pools are also constructed by random sampling.

**Active testing setup and metrics.** We take our loss to be cross–entropy. The rest of our setup is the same as for the regression problems, with the exception that we do not update our surrogate model, in line with the image–classification experiments in [39, 40].

### B.2 Section 5.3

Here, we provide details for the experiments in Figs. 3 and 4.

**The choice of the acquisition strategy.** The regression and classification experiments in Fig. 4 follow the same setup as detailed in §B.1.1 and §B.1.2. The LURE Acq. + PPI approach uses the LURE proposal and the PPI estimator (9)[9]; the Random + PPI approach uses random sampling in combination with the PPI estimator; PPAT uses its standard estimator and acquisition rule. We use $\lambda = 1$ for all the approaches.

**The choice of the estimator.** The regression and classification experiments in Fig. 3 follow the same setup as detailed in §B.1.1 and §B.1.2. The PPAT Acq. + LURE approach uses the PPAT proposal (14) and the LURE estimator (4); the PPAT Acq. + Empirical approach again uses the same PPAT proposal but with the unweighted estimator (2); PPAT uses its standard estimator and acquisition rule. We use $\lambda = 1$ for all the approaches.

### B.3 Section 5.4

Here, we provide details for the coverage experiments in §5.4 (Fig. 5). The regression and classification experiments follow the same setup as detailed in §B.1.1 and §B.1.2. We use an error level of $\delta = 0.1$ and compute the confidence intervals using (25), noting that Random is a special case of LURE with $V_m = 1$, $\forall m$. Coverage is computed with respect to the true test risk on the full pool for each labelling budget. We report the fraction of the 1000 trials for which the confidence interval contains this full–pool test risk.

### B.4 Computational Resources and Licenses

All our experiments were run on a single NVIDIA H100 80GB GPU. Below, we provide the licenses for the packages and datasets used in this work.

Table 1: Software packages used in our experiments.

| Project | Citation | License | URL |
|---|---|---|---|
| `NumPy` | [29] | BSD | `https://numpy.org` |
| `SciPy` | [76] | BSD | `https://scipy.org` |
| `scikit-learn` | [57] | BSD | `https://scikit-learn.org` |
| `Matplotlib` | [31] | PSF–based | `https://matplotlib.org` |
| `pandas` | [56]; [50] | BSD | `https://pandas.pydata.org` |
| `uci_datasets` | [20] | MIT | `https://github.com/treforevans/uci_datasets` |
| `CLIP` | [61] | MIT | `https://github.com/openai/CLIP` |
| `sentence-transformers` | [62] | Apache-2.0 | `https://www.sbert.net` |
| `torchvision` | [47] | BSD | `https://pytorch.org/vision` |
| `laplace-redux` | [17] | MIT | `https://github.com/runame/laplace-redux` |
| `wandb` | [11] | MIT | `https://wandb.ai` |
| `transformers` | [80] | Apache-2.0 | `https://huggingface.co/docs/transformers` |
| `timm` | [79] | Apache-2.0 | `https://huggingface.co/docs/timm` |

[9] Note that we slightly abuse terminology here as (9) is not the PPI estimator described in [3], but instead the unbiased, importance weighted version of it. For random sampling, the two coincide.

Table 2: Datasets used in our experiments.

| Dataset | Citation | License | URL |
|---|---|---|---|
| CIFAR-10 | [41] | Not specified | https://www.cs.toronto.edu/~kriz/cifar.html |
| CIFAR-100 | [41] | Not specified | https://www.cs.toronto.edu/~kriz/cifar.html |
| Tiny-ImageNet | [43] | Not specified | http://cs231n.stanford.edu/tiny-imagenet-200.zip |
| keggdirected | [52] | CC BY 4.0 | https://doi.org/10.24432/C5CK52 |
| keggundirected | [53] | CC BY 4.0 | https://doi.org/10.24432/C5G609 |
| bike | [21] | CC BY 4.0 | https://doi.org/10.24432/C5W894 |
| sml | [65] | CC BY 4.0 | https://doi.org/10.24432/C5RS3S |

# C Additional Results

## C.1 Empirical Verification of $\lambda^\dagger$

Here, we empirically validate the suitability of our target $\lambda^\dagger$. To do this, we compare PPAT run with the fixed value of $\lambda^\dagger$, where $\lambda^\dagger$ is computed on the full pool, with PPAT run with $\lambda \in \{0.1, 0.25, 0.5, 0.75, 1.0, 1.5, 2.5, 5.0\}$. For the experiments, we use one regression dataset (`Keggdirected`) and one classification dataset (`CIFAR-10`), and follow the same setup in §B.1.1 and §B.1.2.

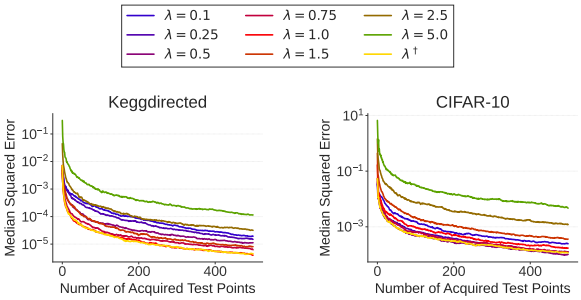



Figure 6: Plot of PPAT for a range of fixed $\lambda$ values and $\lambda^\dagger$ for the datasets `Keggdirected` and `CIFAR-10`. $\lambda^\dagger$ is computed on the full pool. The plots report the median squared error over 1000 trials.

Fig. 6 plots the median squared error across 1000 trials for all the different values of $\lambda$ described previously as well as $\lambda^\dagger$. As we can see, $\lambda^\dagger$ achieves the same median squared error as the best $\lambda$ for both datasets. Besides the discussion in §3.3, this further supports the use of $\lambda^\dagger$ as a suitable $\lambda$ to target instead of the intractable $\lambda^\star$.

## C.2 Additional Ablations

Here, we ablate with different proxy models and different surrogate models.

### C.2.1 Different Proxies

To ablate with different proxy models, we use one regression dataset (`Keggdirected`) and one classification dataset (`CIFAR-10`), and follow the same setup as in §B.1.1 and B.1.2. For the regression dataset, we compare with the `TabPFN-2.0` foundation model, where we make predictions on the pool using in–context learning with the context set being our training set in the same way as before. For the classification case, we compare with the `ViT-B-32` and `ViT-B-16` variants of the CLIP model from the `sentence-transformers` library[10] [62].

From Fig. 7, we see that across both datasets, PPAT remains robust to different choices of the proxy model. Indeed, for all proxy choices, PPAT still retains the lowest median squared error for all choices of $\lambda$.

### C.2.2 Different Surrogate Models

To ablate with different surrogate models, we use one regression dataset (`Keggdirected`) and one classification dataset (`CIFAR-10`), and follow the same setup as in §B.1.1 and B.1.2. For the regression dataset, we use a random forest regression model from `Scikit-Learn` [57] with 300 trees, with the remainder of the hyperparameters at their default values. Similarly, for the classification dataset, we use a random forest classification model from `Scikit-Learn` [57] with 500 trees, with the remainder of the hyperparameters at their default values.

[10] Their corresponding model cards are `ViT-B-32::datacomp_xl_s13b_b90k` and `ViT-B-16::datacomp_xl_s13b_b90k`

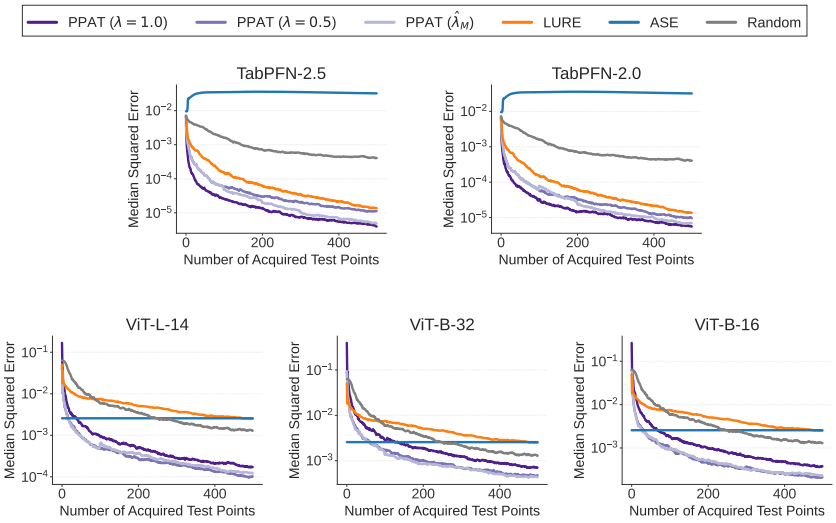



Figure 7: Ablation with different proxy models comparing PPAT with Random, ASE, LURE. **Top row**: results on the `Keggdirected` dataset with `TabPFN-2.5` and `TabPFN-2.0` as the proxies. **Bottom row**: results on the `CIFAR-10` dataset with the `Vit-L-14`, `Vit-B-32`, `Vit-B-16` CLIP models as the proxies. Plots show the median squared error across 1000 trials.

From Fig. 8, we see that across both datasets PPAT remains robust to different choices of surrogate models. Indeed, for all surrogate choices, PPAT still retains the lowest median squared error for all choices of $\lambda$, except for the random forest model on `CIFAR-10` where $\lambda = 1$ performs similarly to LURE.

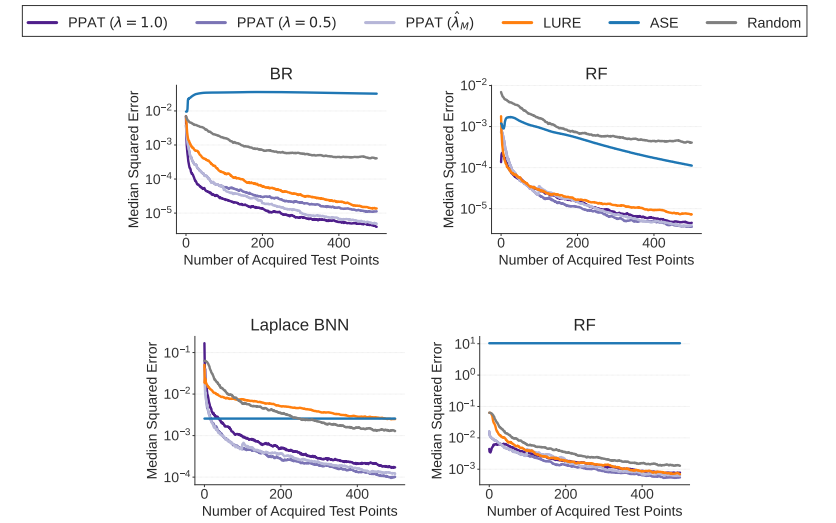



Figure 8: Ablation with different surrogate models comparing PPAT with Random, ASE, LURE. **Top row**: results on the `Keggdirected` dataset with a Bayesian linear regression (BR) and random forest (RF) model as the surrogate. **Bottom row**: results on the `CIFAR-10` dataset with a Laplace BNN and random forest model as the surrogate. Plots show the median squared error across 1000 trials. Note that ASE results in **biased** estimates of the risk.

Moreover, on the `Keggdirected` dataset, ASE outperforms Random when using the random forest surrogate and improves as more labels are acquired. This suggests that the poor behaviour of ASE with the Bayesian linear model surrogate is not only due to active sampling itself, but also due to

surrogate misspecification: because ASE estimates the risk through the surrogate predictive model, errors in this model can translate directly into bias in the risk estimate. The random forest surrogate appears to approximate the losses on this tabular regression task more accurately, reducing this bias enough for the additional acquired labels to improve the estimate. On `CIFAR-10`, however, ASE performs substantially worse than Random when using the random forest surrogate. This suggests that the random forest is not sufficiently expressive for this CLIP–embedding classification task, so the bias introduced by the ASE estimate dominates.

### C.3 Additional Metrics

Here, we present additional metrics for the experiments in §5.

#### C.3.1 Mean Error

Figs. 9, 10, 11, 12 show the mean error (bias) for the experiments in Figs. 1, 2, 3, and 4 respectively. The mean error here is the mean difference between the estimate of the risk and the true test risk on the full pool.

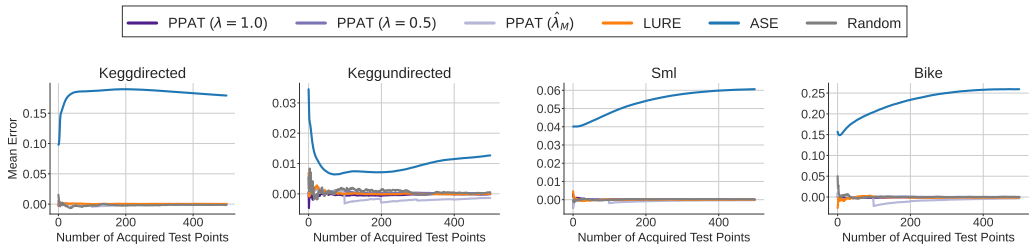



Figure 9: Experiments on UCI datasets comparing PPAT with Random, ASE, LURE. Plots show the mean error (bias) across 1000 trials. Note that ASE results in **biased** estimates of the risk.

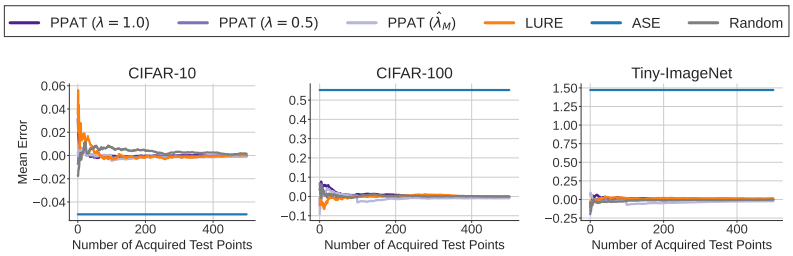



Figure 10: Experiments on classification datasets comparing PPAT with Random, ASE, LURE. Plots show the mean error (bias) across 1000 trials. Note that ASE results in **biased** estimates of the risk.

Across Figs. 9 and 10, ASE exhibits persistent non-zero mean error, as expected: ASE estimates the risk through the surrogate predictive model, and any surrogate misspecification can therefore translate directly into bias in the resulting risk estimate. The plug-in version of PPAT can also introduce finite-sample bias because $\lambda$ is estimated from the actively acquired labels. However, this bias decreases as more labels are acquired, as expected from our theoretical results in §E.5.4.

In Fig. 11, PPAT Acq. + Empirical is biased for the expected reason: the data are selected using a non–uniform active proposal, but the estimator does not apply the LURE importance–weighting correction. Moreover, the ablations PPAT Acq. + LURE in Fig. 11 and LURE Acq. + PPI in Fig. 12, show a bias of approximately zero, which is expected as the estimators are still using the LURE importance–weighting correction.

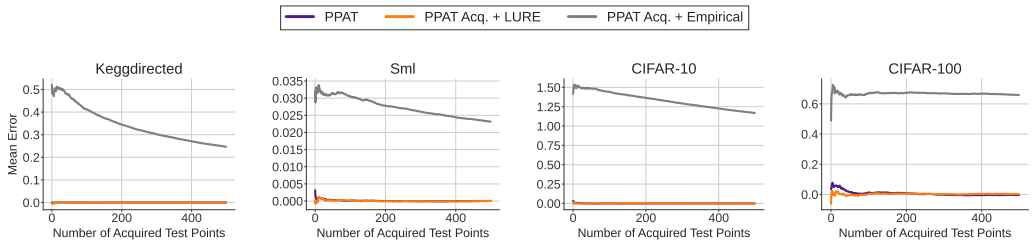



Figure 11: Experiments on UCI and classification datasets studying the influence of our estimator. We compare PPAT with $\lambda = 1$ to using the PPAT acquisition strategy, $Q^{\text{PPAT}}$, but with the LURE estimate of the risk (PPAT Acq. + LURE) and the empirical estimate of the risk (PPAT Acq. + Empirical). Plots show the mean error (bias) across 1000 trials. Note that PPAT Acq. + Empirical results in **biased** estimates of the risk as the data is not randomly sampled.

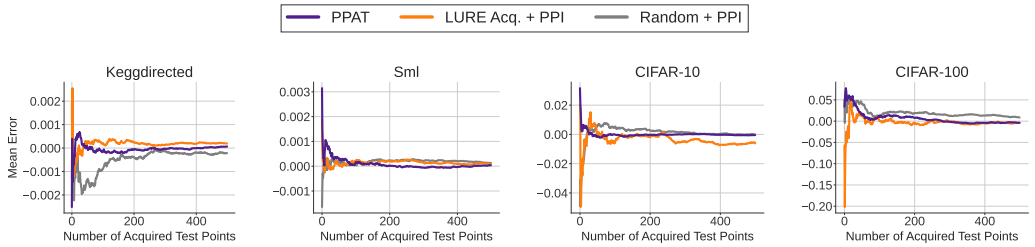



Figure 12: Experiments on UCI and classification datasets studying the influence of our acquisition strategy. We compare PPAT with $\lambda = 1$ to using the PPI estimator with the LURE acquisition strategy (LURE Acq. + PPI) and random sampling (Random + PPI). Plots show the mean error (bias) across 1000 trials.

### C.3.2 Confidence Interval Widths

Fig. 13 shows that the faster attainment of the target coverage level in Fig. 5 is not simply due to overly conservative confidence intervals. Across representative datasets, we see that PPAT achieves lower mean widths than competing approaches, while reaching the desired coverage level at a similar or substantially faster rate. The main exception is `CIFAR-100`, where PPAT with $\lambda = 0.5$ performs similarly to LURE in terms of mean width. Overall, this suggests that the prediction–powered control variate improves not only point estimation, but also (frequentist) uncertainty quantification.

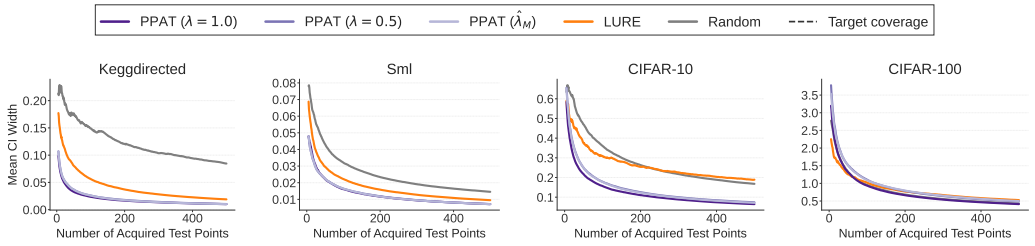



Figure 13: Representative experiments on UCI and classification datasets for the mean widths of the confidence intervals of PPAT, Random, LURE. Plots show the widths of asymptotic confidence intervals constructed with $\delta = 0.1$ across 1000 trials. The ASE baseline is excluded as it does **not** provide asymptotically valid confidence intervals.

## C.4 Additional Comparisons

Here, we present a broader set of comparative analyses against PPAT.

### C.4.1 Further Comparisons with Prediction–Powered Inference

Fig. 14 extends Fig. 4 by additionally including the standard Random and LURE baselines, shown as dashed lines. Across all datasets, Random + PPI outperforms both Random and LURE, further demonstrating that prediction–powered inference provides a strong baseline. Nevertheless, PPAT consistently achieves lower median squared error. This shows that the gains of PPAT do not come only from using a PPI–like estimator, but also from adaptively sampling points in a way that is tailored to reducing the variance of that estimator.

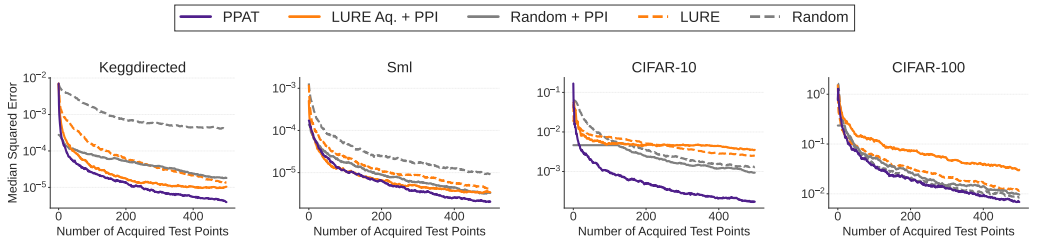



Figure 14: Experiments on UCI and classification datasets studying the influence of our acquisition strategy. We compare PPAT with $\lambda = 1$ to using the PPI estimator with the LURE acquisition strategy (LURE Acq. + PPI, ——) and random sampling (Random + PPI, ——), as well as the standard LURE (- - -) and Random (- - -) baselines. Plots show the median squared error across 1000 trials.

### C.4.2 Comparison with Active Surrogate Estimators Using the Proxy

In §5, we compared PPAT with the original formulation of ASE. Here, we consider a related baseline that uses the proxy predictions directly. Specifically, since the proxy $g$ is evaluated on the full test pool, we form the proxy losses $\widetilde{\ell}_i$ and estimate the risk by their full-pool average. We refer to this baseline as Proxy. This tests whether the initial proxy model is already accurate enough to replace the unknown labels, rather than using the proxy only as a control variate as in PPAT.

This baseline is related to ASE, but is not directly comparable to the original ASE approach. In our setting, the proxy $g$ can be a fixed black–box predictor: even if its predictions are available on the entire pool, we may not have access to posterior samples, ensemble members, or other measures of epistemic uncertainty needed to compute the XWED acquisition score. Like ASE, however, Proxy generally yields a biased estimate of the risk and does not inherit the unbiasedness guarantee of PPAT.

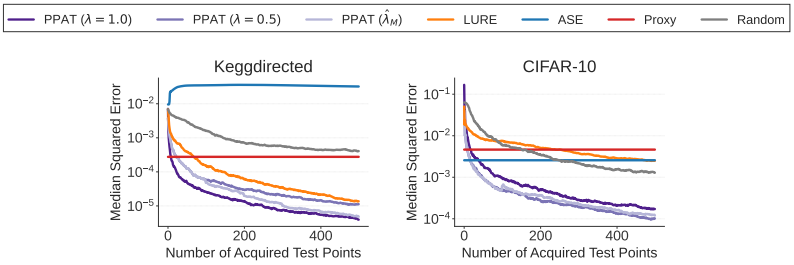



Figure 15: Experiments on `Keggdirected` and `CIFAR-10` comparing PPAT with Random, ASE, Proxy, LURE. Plots show the median squared error across 1000 trials. Note that ASE and Proxy result in **biased** estimates of the risk.

We report this comparison on `Keggdirected` and `CIFAR-10`, representing one regression dataset and one classification dataset. We use the same setup as in Figs. 1 and 2 which is described in detail in §B.1.1 and §B.1.2 respectively.

Fig. 15 shows that PPAT continues to outperform the Proxy baseline for all choices of $\lambda$ on both datasets. This indicates that, although the proxy predictions contain useful information, directly treating them as labels is not sufficient to match the performance of our approach. Interestingly,

the Proxy baseline performs better than ASE on the regression dataset, but worse than ASE on the classification dataset. A likely explanation is that, in the regression setting, the `TabPFN-2.5` proxy makes predictions using in–context learning with the task–specific training set, making its proxy losses relatively informative for the test pool. In contrast, in the classification setting, the CLIP proxy predictions are zero–shot and are not adapted to the specific downstream classification task, so the resulting proxy losses may be less aligned with the true losses.

### C.4.3 Comparison with Active Statistical Inference

Here, we compare PPAT with Active Statistical Inference (ASI, [84]). Like before, we use one regression (`Keggdirected`) and one classification dataset (`CIFAR-10`). For a fair comparison with PPAT, we use the "batch" setting of ASI for `CIFAR-10` and the "active" setting of ASI for `Keggdirected`, where we use the default hyperparameters from the paper for both[11]. Note that for the "active" setting of ASI, the proxy model is updated periodically as we collect more labels, whereas for us the proxy model is kept fixed.

Table 3: Median squared error, mean confidence–interval width, and mean coverage on `Keggdirected` and `CIFAR-10` over 1000 trials, reported to two significant figures. $\pm$ indicates one standard error; standard errors below $1 \times 10^{-3}$ are reported as 0.00. Note that ASE does not provide confidence intervals, so its width and coverage results are omitted.

| Method | `Keggdirected` | | | `CIFAR-10` | | |
|---|---|---|---|---|---|---|
| | Med. squared error | Mean width | Mean coverage | Med. squared error | Mean width | Mean coverage |
| ASE [40] | $3.2 \times 10^{-2}{\pm 0.00}$ | — | — | $2.6 \times 10^{-3}{\pm 0.00}$ | — | — |
| LURE [39] | $1.4 \times 10^{-5}{\pm 0.00}$ | $1.9 \times 10^{-2}{\pm 0.00}$ | $0.89{\pm 0.0099}$ | $2.5 \times 10^{-3}{\pm 0.00}$ | $1.9 \times 10^{-1}{\pm 0.0047}$ | $0.68{\pm 0.015}$ |
| Random | $4.1 \times 10^{-4}{\pm 0.00}$ | $7.2 \times 10^{-1}{\pm 0.013}$ | $0.084{\pm 0.0018}$ | $1.3 \times 10^{-3}{\pm 0.00}$ | $1.7 \times 10^{-1}{\pm 0.0011}$ | $0.83{\pm 0.010}$ |
| ASI [84] | $4.5 \times 10^{-5}{\pm 0.00}$ | $4.9 \times 10^{-2}{\pm 0.00}$ | $0.99{\pm 0.0024}$ | $9.8 \times 10^{-5}{\pm 0.00}$ | $6.5 \times 10^{-2}{\pm 0.0010}$ | $0.98{\pm 0.0042}$ |
| PPAT ($\lambda = 1$) | $4.0 \times 10^{-6}{\pm 0.00}$ | $1.0 \times 10^{-2}{\pm 0.00}$ | $0.90{\pm 0.0096}$ | $1.7 \times 10^{-4}{\pm 0.00}$ | $6.5 \times 10^{-2}{\pm 0.00}$ | $0.90{\pm 0.0097}$ |
| PPAT ($\lambda = 0.5$) | $1.1 \times 10^{-5}{\pm 0.00}$ | $1.0 \times 10^{-2}{\pm 0.00}$ | $0.89{\pm 0.010}$ | $1.0 \times 10^{-4}{\pm 0.00}$ | $7.4 \times 10^{-2}{\pm 0.00}$ | $0.90{\pm 0.0097}$ |
| PPAT ($\lambda = \widehat{\lambda}_M$) | $4.8 \times 10^{-6}{\pm 0.00}$ | $1.0 \times 10^{-2}{\pm 0.00}$ | $0.88{\pm 0.010}$ | $1.2 \times 10^{-4}{\pm 0.00}$ | $7.1 \times 10^{-2}{\pm 0.00}$ | $0.89{\pm 0.010}$ |

Following the original ASI paper, we report the final mean coverage and widths of the confidence intervals of both approaches, as well as the median squared error. From Table 3 we see that on `Keggdirected`, PPAT improves over ASI in both risk estimation and confidence–interval width while still attaining the desired coverage level: each of the three PPAT variants attains a lower median squared error and a substantially narrower mean interval width than ASI, whose intervals are moreover conservative relative to the nominal level. We believe this advantage stems from the pool–based nature of our acquisition strategy: because PPAT conditions on the full test pool, each acquisition is guided not only by the previously acquired labels but also by the remaining pool, allowing it to use more information when deciding which points to query. By contrast, the active variant of ASI used here acquires labels in a more streaming manner. On `CIFAR-10`, however, ASI performs comparably to, or slightly better than, PPAT in terms of median squared error and mean interval width, while PPAT attains coverage closer to the nominal level. This is unsurprising, since the batch variant of ASI used here is closely related to our approach: both are ultimately built on importance–weighted estimates of the risk but with different proposals.

## C.5 On Variance Reduction over LURE

Here, we further expand on the variance reduction of PPAT over LURE stated in Prop. 3.2. Throughout this section, $Q$ denotes the acquisition process, including its potentially adaptive sequence of proposals. We use $\rho_Q$ to denote the correlation between $\widehat{R}_{\mathrm{LURE}}$ and $\widehat{C}_{\mathrm{LURE}}$ over acquisition randomness, conditional on the fixed test pool, predictor, and proxy. We also use $\rho$ to denote the correlation between the true and proxy losses over the full test pool:

$$\rho = \mathrm{Corr}(\ell_J, \widetilde{\ell}_J), \qquad J \sim \mathrm{Unif}(\{1, \dots, N\}).$$

Firstly, recall that the variance reduction of PPAT over LURE in Prop. 3.2 requires both estimators to use the same proposals. In practice, however, we recommend using proposals tailored to their

[11] This includes the mixing hyperparameter $\tau$ and the finetuning batch size. We did not find any noticeable gains from further tuning of the hyperparameters.

respective estimators (see §5.3). Moreover, we know that the largest variance reduction occurs at $\lambda^\star$ where the variance is reduced by a factor of $(1 - \rho_Q^2)$. While helpful, these results do not by themselves guarantee variance reduction when PPAT uses the proposal in (14) and the LURE baseline uses (6). The correlation $\rho$ is also generally not easily interpretable as it is the correlation between two importance–weighted estimators.

To better understand when we can expect PPAT to achieve lower variance than LURE in practice, we ask the following two questions: *i) Can $\rho_Q$ provide any insight into the amount of variance reduction achieved by PPAT over LURE when using (6), for both our fixed and plug–in choices of $\lambda$? ii) Can the more easily interpretable $\rho$ give any insight into the behaviour of $\rho_Q$?* We answer both of these questions through a controlled, synthetic experiment where we can easily adjust $\rho$ by changing our proxy.

**Experimental setup.** For our experiments, we consider a regression task and use the squared–error loss to estimate the risk of $f$ on a synthetic test pool. To generate our test pool, we sample 2,250 covariates, each with 6 features, using $\mathbf{X}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{6\times 6})$, and sample our response using

$$h(\mathbf{x}) = \beta^\top \mathbf{x} + 0.8\sin(2x_1) + 0.6x_2x_3 - 0.5x_4^2$$
$$Y_i = h(\mathbf{X}_i) + \epsilon_i, \quad \epsilon_i \sim \mathcal{N}(0, 0.2^2)$$

where $\boldsymbol{\beta} \sim \mathcal{N}(0, \mathbf{I}_{6\times 6}/6)$ and $\mathbf{x} = [x_1 \cdots x_6]^\top$. We use the first 250 points for our training set and the remainder for our test pool.

We use a ridge regression model for $f$, fit on our training set using only the first three features, so that $f$ cannot fully recover $h$ and thus mirrors the practical setting of evaluating an imperfect predictor. For our surrogate, we use a Bayesian linear regression model fit on our training set using all of the features. We update the surrogate after each acquired label from the pool.

We define our proxy as

$$g_\tau(\mathbf{x}) = (1 - \tau)h(\mathbf{x}) + \tau q(\mathbf{x})$$

which interpolates between the true, underlying signal $h(\mathbf{x})$ and the smooth function

$$q(\mathbf{x}) = \mathbf{v}^\top \mathbf{x} + 0.7\cos(1.5x_2) - 0.6x_5^2 + 0.5x_1x_4,$$

where $\mathbf{v} \sim \mathcal{N}(0, \mathbf{I}_{6\times 6}/6)$ and $\tau \in [0, 1]$. By choosing the proxy in this way, we are able to vary the correlation $\rho$ by choosing $\tau$ from a grid of values in $[0, 1]$ such that the resulting correlation is as close as possible to our desired correlation level. We consider the following correlation levels: $0.90, 0.80, 0.75, 0.50, 0.40, 0.25, 0.10$.

We use a labelling budget of $M = 150$ and consider $\lambda = 0.5, 1, \widehat{\lambda}$. For each proxy, we run 100 independent trials while keeping the training set, test pool, predictor $f$, and proxy $g_\tau$ fixed.

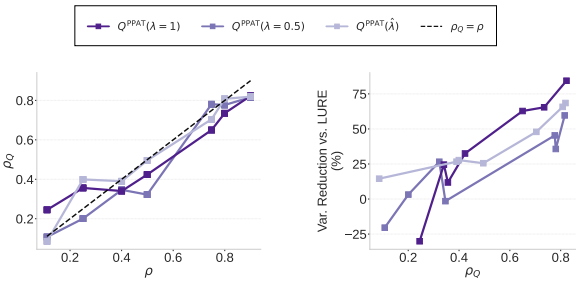



Figure 16: Plots of $\rho_Q$ vs. $\rho$ for PPAT using $\lambda = 1, 0.5$, and $\widehat{\lambda}$ (left), and the mean variance reduction of PPAT over LURE, based on the proposal from (6), plotted against $\rho_Q$ (right). All results are reported across 100 trials in our synthetic setting.

**Does a higher $\rho_Q$ lead to greater variance reduction?** Figure 16 shows the mean variance reduction achieved by PPAT over LURE at the final labelling budget against $\widehat{\rho}_Q$. Here, we see that a larger $\rho_Q$ generally results in a larger variance reduction over LURE at all the $\lambda$ values considered,

suggesting that $\rho_Q$ can serve as an indicator of the amount of variance reduction that we can expect to observe over LURE beyond $\lambda^\star$.

To investigate whether this behaviour holds more generally, we also consider using different proposals with the PPAT estimator. In particular, we also use the LURE proposal in (6), Random sampling, and two new proposals, $|c|$and $|c|^2$, which sample points from the pool proportionally to $|c|$ and $|c|^2$, respectively. While we do not use these last two proposals elsewhere in the paper, they allow us to develop a deeper understanding of the relationship between $\rho_Q$ and the reduction in variance that PPAT can achieve over LURE.

As can be seen from Figure 17, the same behaviour as before generally holds for other proposals as well. However, we note that the behaviour is strongest for Random and LURE, and less clear for the other two proposals. We hypothesize that this is because $|c|$and $|c|^2$do not directly target quantities that are relevant to our estimator – they target the raw control variates as opposed to the residualised losses which appear in our estimator. This is further supported by the behaviour being even less strong for $|c|^2$, which more aggressively targets the raw control variates.

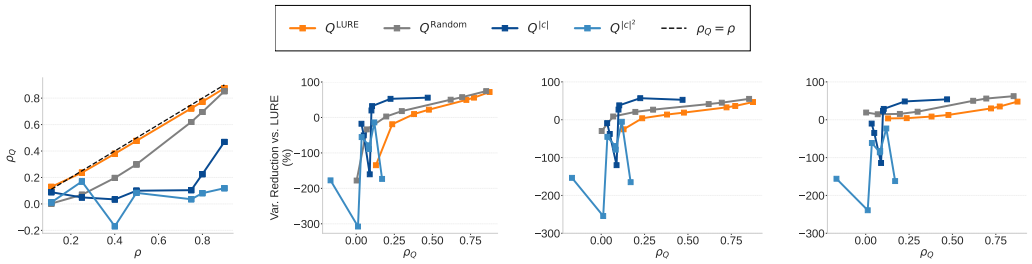



Figure 17: Plots of $\rho_Q$ vs. $\rho$ for the proposals LURE, Random, $|c|$, and $|c|^2$(leftmost). The remaining three plots display the mean variance reduction of PPAT with these proposals over LURE vs. $\rho_Q$, when using $\lambda = 1, 0.5$, and $\widehat{\lambda}$ in the estimator, respectively. All plots report results across 100 trials in our synthetic setting.

**How does $\rho$ relate to $\rho_Q$?** Figure 16 plots $\rho_Q$ as a function of $\rho$ for PPAT. Here, we see that $\rho$ correlates relatively strongly with $\rho_Q$, suggesting that an increase in $\rho$ can serve as a reliable proxy for $\rho_Q$. Moreover, as before, we also see from Figure 17 that this behaviour can hold more generally for other proposals, such as LURE and Random[12], but does not hold for $|c|$and $|c|^2$. This suggests that the choice of sampling strategy can be crucial for determining whether this behaviour actually holds in practice.

Returning to our original questions, the two observations above tell us that an increase in $\rho_Q$ can indicate greater variance reduction over LURE beyond $\lambda^\star$, and similarly, that a higher $\rho$ can indicate a larger $\rho_Q$. While these observations hold for our PPAT proposals, the results also suggest that these behaviours can vary depending on the proposal.

**Remark C.1.** While we would ideally like to find a rigorous mathematical connection between $\rho$ and $\rho_Q$, and between $\rho_Q$ and the variance reduction over LURE, we have found that in practice, this is not possible without imposing additional, restrictive assumptions.

[12]For Random, which samples uniformly without replacement, $\rho_Q = \rho$ exactly conditional on the fixed test pool. Empirical estimates of $\rho_Q$ based on finitely many sampling trials can deviate from the black dotted identity line because of Monte Carlo estimation error.

# D Additional Details and Discussions

## D.1 On the Choice of the Surrogate Score in Section 2.1

Here, we discuss the choice of the surrogate score in §2.1. This also explains our choice in §3.2.

The myopic oracle proposal of §2.1 is given by

$$Q_m^\star(i) \;\propto\; \big|\ell_i\big|. \tag{18}$$

Because the true label $y_i$ (and hence $\ell_i$) is unknown for $i \in S_m$, $|\ell_i|$ must be replaced by a surrogate score computed from the surrogate predictive distribution $\pi_m(\cdot \mid \mathbf{x}_i)$. There are two natural choices:

$$a_m^{\mathrm{abs}}(i) := \mathbb{E}_{Y\sim\pi_m(\cdot|\mathbf{x}_i)}\big[\,\big|\mathcal{L}(f(\mathbf{x}_i),Y)\big|\,\big], \tag{19}$$

$$a_m(i) := \sqrt{\mathbb{E}_{Y\sim\pi_m(\cdot|\mathbf{x}_i)}\big[\,\mathcal{L}(f(\mathbf{x}_i),Y)^2\,\big]}\,. \tag{20}$$

The score $a_m^{\mathrm{abs}}$ is obtained by applying $\pi_m$ directly to the oracle expression $|\ell_i|$; this is the score that one might expect on first reading. The score $a_m$ in (20) is the one we use, and it coincides with $a_m^{\mathrm{abs}}$ only when $\mathcal{L}(f(\mathbf{x}_i),Y)$ is constant under $\pi_m(\cdot \mid \mathbf{x}_i)$. Below, we discuss why $a_m$ is the more principled surrogate of the oracle.

Firstly, note that the oracle proposal $Q_m^\star$ in (18) is derived by minimising the conditional sampling variance of LURE, which is upper–bounded by the second-moment objective (see §E.4 for further details)

$$\mathbb{E}\left[\left(\frac{\ell_{I_m}}{N\,Q_m(I_m)}\right)^2 \;\middle|\; I_{1:m-1}\right] \;=\; \frac{1}{N^2}\sum_{i\in S_m}\frac{\ell_i^2}{Q_m(i)}. \tag{21}$$

Now, note that the objective on the right–hand side is *quadratic* in the loss: the per–point contribution is $\ell_i^2/Q_m(i)$, not $|\ell_i|/Q_m(i)$. The closed–form oracle $Q_m^\star(i) \propto |\ell_i|$ is obtained by solving this quadratic problem in closed form via Lagrange multipliers, meaning that the absolute value is a consequence of the optimisation, not of the objective.

When $\ell_i$ is unknown and a surrogate $\pi_m$ is introduced, the right quantity to substitute into (21) is therefore the predictive expectation of $\ell_i^2$, i.e. the second moment, not a point estimate of $|\ell_i|$. Concretely, replacing $\ell_i^2$ by $\mathbb{E}_{\pi_m(\cdot|\mathbf{x_i})}[\mathcal{L}(f(\mathbf{x}_i),Y)^2]$ in (21) and re–solving the same minimisation problem yields exactly

$$Q_m^{\mathrm{surr}}(i) \;\propto\; \sqrt{\mathbb{E}_{\pi_m(\cdot|\mathbf{x_i})}\big[\mathcal{L}(f(\mathbf{x}_i),Y)^2\big]} \;=\; a_m(i).$$

In other words, $a_m$ is the surrogate score induced by the *variance objective itself*, not by a post–hoc substitution into the closed–form oracle.

## D.2 Simplified Expressions for the Proposals of LURE and PPAT

Here, we describe the proposals used for LURE and PPAT in our experiments. We discuss the case for the MSE loss – the case for the cross–entropy loss follows similarly.

**LURE.** Let $f_i := f(\mathbf{x}_i)$, $\mu_{m,i} := \mathbb{E}_{\pi_m(\cdot|\mathbf{x}_i)}[y]$ and $\sigma_{m,i}^2 := \mathrm{Var}_{\pi_m(\cdot|\mathbf{x}_i)}(y)$. For the MSE loss, we follow [39] and use the proposal

$$Q_m(i) \propto \mathbb{E}_{\pi_m(\cdot|\mathbf{x}_i)}\big[(f_i-y)^2\big] = \mathrm{Var}_{\pi_m(\cdot|\mathbf{x}_i)}(y) + \Big(\mathbb{E}_{\pi_m(\cdot|\mathbf{x}_i)}[y]-f_i\Big)^2 = \sigma_{m,i}^2 + (\mu_{m,i}-f_i)^2.$$

Note that this is not the same as the proposal $Q_m(i) \propto \sqrt{\mathbb{E}_{\pi_m(\cdot|\mathbf{x}_i)}[(f_i-y)^4]}$ described in §2.1 and §D.1.

From the above, we can see that under the MSE loss the LURE proposal has the standard bias–variance decomposition form: it favours points for which either the surrogate is uncertain about the label, or the surrogate mean disagrees strongly with the prediction of the model being evaluated.

**PPAT.** Let $c_i = \widetilde{\ell}_i - \widetilde{R}$, and $\widetilde{\ell}_i = (f_i - g(\mathbf{x}_i))^2$. The proposal $a_{m,\lambda}$ (13) admits the following decomposition

$$\mathbb{E}_{\pi_m(\cdot|\mathbf{x}_i)}\Big[\big(\mathcal{L}(f(\mathbf{x}_i), Y) - \lambda c_i\big)^2\Big] = \mathrm{Var}_{\pi_m(\cdot|\mathbf{x}_i)}(\mathcal{L}(f(\mathbf{x}_i), Y)) + \big(\mathbb{E}_{\pi_m(\cdot|x_i)}[L(f(x_i), Y)] - \lambda c_i\big)^2. \tag{22}$$

This shows that the PPAT proposal favours points for which either the surrogate is uncertain about the loss, or the surrogate mean loss is far from the proxy term $\lambda c_i$.

Under the MSE loss, (22) becomes

$$\mathbb{E}_{\pi_m(\cdot|\mathbf{x}_i)}\Big[\big((f_i - Y)^2 - \lambda c_i\big)^2\Big] = \mathrm{Var}_{\pi_m(\cdot|\mathbf{x}_i)}\big((f_i - Y)^2\big) + \big(\mathbb{E}_{\pi_m(\cdot|\mathbf{x}_i)}[(f_i - Y)^2] - \lambda c_i\big)^2.$$

Now, using the standard bias–variance decomposition, the second term simplifies to

$$\big(\sigma^2_{m,i} + (\mu_{m,i} - f_i)^2 - \lambda c_i\big)^2.$$

To simplify the first term, let $a_{m,i} = f_i - \mu_{m,i}$ and $z = Y - \mu_{m,i}$ so that $\mathbb{E}[z] = 0$, $\mathrm{Var}(z) = \sigma^2_{m,i}$, and $(f_i - Y)^2 = (a_{m,i} - z)^2$. Then we have:

$$\mathrm{Var}\big((f_i - Y)^2\big) = \mathrm{Var}(z^2 - 2a_{m,i}z) = \big(\mathbb{E}[z^4] - \sigma^4_{m,i}\big) + 4a^2_{m,i}\sigma^2_{m,i} - 4a_{m,i}\mathbb{E}[z^3].$$

Now, assuming the surrogate predictive distribution is Gaussian[13], i.e. $Y \mid \mathbf{x}_i \sim \mathcal{N}(\mu_{m,i}, \sigma^2_{m,i})$, then $\mathbb{E}[z^3] = 0$ and $\mathbb{E}[z^4] = 3\sigma^4_{m,i}$, giving

$$\mathrm{Var}_{\pi_m(\cdot|\mathbf{x}_i)}\big((f_i - Y)^2\big) = 2\sigma^4_{m,i} + 4\sigma^2_{m,i}(\mu_{m,i} - f_i)^2.$$

Therefore, under the MSE loss and a Gaussian surrogate, the PPAT proposal simplifies to

$$Q^{\mathrm{PPAT}}_{m,\lambda}(i) \propto \sqrt{2\sigma^4_{m,i} + 4\sigma^2_{m,i}(\mu_{m,i} - f_i)^2 + \big(\sigma^2_{m,i} + (\mu_{m,i} - f_i)^2 - \lambda c_i\big)^2}.$$

### D.3 Extended Background

#### D.3.1 Active Testing with Active Surrogate Estimators

Here, we provide further details on the XWED acquisition function used in [40].

At acquisition round $m$, let $\pi_m(\cdot \mid \mathbf{x})$ denote the surrogate predictive distribution used in the main text. When this surrogate is Bayesian, ensemble-based, or otherwise represented as a mixture, we write $\phi_m$ for the fitted quantities that determine the surrogate at round $m$ – for example, posterior hyperparameters, variational parameters, or the trained ensemble – and let $\Theta$ denote the latent component of the surrogate predictive model. With this notation, the round-$m$ surrogate predictive distribution can be written as the posterior predictive mixture

$$\pi_m(Y \mid \mathbf{x}) \equiv \pi_m(Y \mid \mathbf{x}; \phi_m) = \mathbb{E}_{\Theta \sim \pi_m(\cdot;\phi_m)}\big[\pi_m(Y \mid \mathbf{x}, \Theta; \phi_m)\big].$$

The XWED score is then given by

$$\begin{aligned}\mathrm{XWED}(\mathbf{x}) = {} & \mathbb{E}_{Y \sim \pi_m(\cdot|\mathbf{x})}[-\mathcal{L}(f(\mathbf{x}), Y) \log \pi_m(Y \mid \mathbf{x})] \\ & - \mathbb{E}_{\Theta \sim \pi_m(\cdot;\phi_m)}\big[\mathbb{E}_{Y \sim \pi_m(\cdot|\mathbf{x},\Theta;\phi_m)}[-\mathcal{L}(f(\mathbf{x}), Y) \log \pi_m(Y \mid \mathbf{x}, \Theta; \phi_m)]\big].\end{aligned}$$

with the following acquisition rule at round $m$:

$$i_m = \arg\max_{i \in S_m} \mathrm{XWED}(\mathbf{x}_i).$$

[13] This is the case for our experiments where we use a Bayesian linear regression model with Gaussian priors as the surrogate. It is also the case for a large family of models such as Gaussian process regression with a Gaussian likelihood and Bayesian linear regression with conjugate Gaussian priors. For non–Gaussian surrogates, $\mathbb{E}[z^3]$ and $\mathbb{E}[z^4]$ may not be available in closed form, in which case the expectation in (13) can instead be estimated by a Monte Carlo average of $\big((f_i - Y)^2 - \lambda c_i\big)^2$ over i.i.d. samples $Y \sim \pi_m(\cdot \mid x_i)$.

### D.4 Future Work and Limitations

Here, we discuss the limitations of our approach and possible directions for future work.

**Future work.** First, note that $R$ in eq. (8) can also be written as $\lambda\widetilde{R} + \frac{1}{N}\sum_{i=1}^{N}(\ell_i - \lambda\widetilde{\ell}_i)$. Applying LURE to this expression, and noting that $\widetilde{R}$ is constant, we arrive at the alternative PPAT estimator:

$$\lambda\widetilde{R} + \frac{1}{M}\sum_{m=1}^{M} V_m(\ell_{I_m} - \lambda\tilde{\ell}_{I_m}). \tag{23}$$

This generally differs from eq. (9) because the LURE weights may not sum to $M$; however, both estimators are unbiased for a fixed $\lambda$. In our work, we use the estimator in eq. (9) for simplicity. However, It would be interesting to explore how this alternative estimator compares in future work.

Moreover, while we use separate proxy and surrogate models and keep the proxy predictions fixed in this paper, our approach can naturally be extended to update the proxy as new labels are acquired. It is also possible to use a single probabilistic model to serve both roles. Keeping the models separate, however, allows us to use a compact surrogate that is computationally cheaper to update than fine-tuning a large predictor (such as a foundation model), while also accommodating proxies accessible only as black boxes. Additionally, it is often more practical to train a well-calibrated model – which is crucial for active testing surrogates (see also §5.1) – directly on labels acquired from the test pool, rather than fine-tuning a model pretrained on potentially different data distributions.

**Limitations.** Our approach also suffers from the same limitations as existing LURE-based active testing works. In particular, our surrogate models must be well-calibrated, which requires not only proper model calibration but also a potentially non-trivial amount of data to train the surrogate. Without this, our importance weights can become unstable and actually result in worse performance than simple random sampling. However, we note that this limitation can be mitigated by using existing foundation models as our surrogates, which have already demonstrated strong performance and calibration across different domains and tasks [18, 51, 75].

Additionally, while we have shown that our approach can still yield gains with proxies of different strengths, we have found that it generally achieves the largest gains with proxies that demonstrate moderate to strong performance on our task.

# E Mathematical Results

Here, we present the full statements and proofs for the results in the main body of our paper. Since the main body states results in a simplified form for readability, we first introduce the definitions needed to state these results formally in §E.1. All subsequent results in this section make use of these definitions.

We note that our finite-sample unbiasedness, variance, and $L^2$-consistency arguments (§E.2–E.5.1) closely follow the analysis of Farquhar et al. [22, Appendix B.4–B.7]. The main difference is that we state the arguments conditionally on a fixed finite test pool and for general real-valued arrays $\xi_{M,i}$, which generalises the results in [22] to include both standard LURE ($\xi_{M,i} = \ell_i$) and PPAT ($\xi_{M,i} = \ell_i - \lambda(\widetilde{\ell}_i - \tilde{R}_M)$). In addition to this, §E.5.2 provides a new asymptotic normality result for the resulting LURE-style estimators.

## E.1 Definitions

We first introduce the definition of an *active proposal process* (APP), which formalises the adaptive sampling proposals considered throughout the paper.

**Definition E.1** (Active proposal process, APP)**.** Let $H_0$ denote all information available before querying the true labels. This includes the inputs $\mathbf{x}_{1:N}$, model predictions, or other side information, but it does not include the unqueried true labels.

An admissible active proposal process (APP) $Q_{1:M} = (Q_1, \dots, Q_M)$ is generated by a collection of measurable maps $\eta_1, \dots, \eta_M$ such that, at step $m$, the map $\eta_m$ takes as input the observed history $H_0, (I_1, y_{I_1}), \dots, (I_{m-1}, y_{I_{m-1}})$ and returns a probability distribution on $S_m$ with full support. The proposal used at step $m$ is denoted by

$$Q_m = \eta_m\big(H_0, (I_1, y_{I_1}), \dots, (I_{m-1}, y_{I_{m-1}})\big)\,,$$

and satisfies

$$Q_m(i) = 0 \quad \text{for } i \notin S_m, \qquad \sum_{i \in S_m} Q_m(i) = 1, \qquad Q_m(i) > 0 \quad \text{for } i \in S_m.$$

The next queried index is sampled according to

$$\Pr\big(I_m = i \mid H_0, I_1, y_{I_1}, \dots, I_{m-1}, y_{I_{m-1}}\big) = Q_m(i), \qquad i \in S_m.$$

We now formalise the definition of a surrogate model introduced in §2.1.

**Definition E.2** (Surrogate model)**.** Let $\mathcal{F}_m = \sigma(\{I_1, \dots, I_m\})$. At round $m$, a surrogate model is an $\mathcal{F}_{m-1}$–measurable Markov kernel $\pi_m(\cdot \mid x)$ on $\mathcal{Y}$. That is, for each input $x$, $\pi_m(\cdot \mid x)$ is a predictive distribution over the unknown label, constructed using only information available before round $m$.

Typical examples include Bayesian neural networks, random forests and Gaussian processes.

## E.2 Generalising LURE with Triangular–Arrays

We begin by introducing a general triangular–array formulation that captures both the standard LURE estimator and the PPAT estimator as special cases, and we establish two algebraic identities – a telescoping representation in terms of running averages and a normalisation identity for the finite–pool weights – that underpin all subsequent results.

Let $(\mathbf{x}_1, \mathbf{x}_2, \dots)$ be a sequence of inputs, with associated labels $(y_1, y_2, \dots)$. Moreover, let $(N_1, N_2, \dots)$ be a sequence of test pool sizes, with $N_M/M \to \alpha$ as $M \to \infty$ for some $\alpha > 1$, and let $(\xi_{M,i})_{M \geq 1, i=1,\dots,N_M}$ be a triangular array of reals, with $\xi_{M,i} = h_M(\mathbf{x}_{1:N_M}, y_i)$ for some function $h_M : \mathcal{X}^{N_M} \times \mathcal{Y} \to \mathbb{R}$. Define $R_M$ as

$$R_M := \frac{1}{N_M} \sum_{i=1}^{N_M} \xi_{M,i}.$$

**Remark E.3.** *In this paper, we consider either $\xi_{M,i} = \ell_i$ (standard LURE), where $\ell_i = \mathcal{L}(f(\mathbf{x}_i), y_i)$, or $\xi_{M,i} = \ell_i - \lambda(\widetilde{\ell}_i - \widetilde{R}_M)$ (PPAT), where $\widetilde{\ell}_i = \mathcal{L}(f(\mathbf{x}_i), g(\mathbf{x}_i))$ and $\widetilde{R}_M = \frac{1}{N_M}\sum_{i=1}^{N_M} \widetilde{\ell}_i$. Note that in both cases, $R_M = \frac{1}{N_M}\sum_{i=1}^{N_M} \ell_i$.*

Consider the LURE estimator of $R_M$ given by

$$\widehat{R}_M = \frac{1}{M}\sum_{m=1}^{M} V_{M,m}\xi_{M,I_{M,m}},$$

with $I_{M,m} \mid I_{M,1:m-1} \sim Q_{M,m}$, where, for each $M$, $Q_{M,1},\dots,Q_{M,M}$ is a sequence of active proposals in the sense of Def. E.1, and

$$V_{M,m} = 1 + \frac{N_M - M}{N_M - m}\left(\frac{1}{(N_M - m + 1)\, Q_{M,m}(I_{M,m})} - 1\right). \tag{24}$$

Throughout this section, for each $M$, we treat the finite test pool $(\mathbf{x}_1, y_1),\dots,(\mathbf{x}_{N_M}, y_{N_M})$ as fixed. All probabilities, expectations, variances, and asymptotic statements are with respect to the randomisation of the active sampling procedure only. Moreover, we write $N$ for $N_M$ whenever the dependence on $M$ is clear and let $S_{M,m} = [N] \setminus \{I_{M,1},\dots,I_{M,m-1}\}$ be the set of unqueried indices before round $m$, and $\mathcal{F}_{M,m} = \sigma(I_{M,1},\dots,I_{M,m})$.

We first state a telescoping representation of LURE in terms of running averages.

**Lemma E.4** (Alternative representation of LURE)**.** *For each $M$, define*

$$\gamma_{M,m} = \frac{N(N-M)}{(N-m)(N-m+1)}$$

*and*

$$A_{M,m} = \frac{1}{N}\left\{\frac{\xi_{M,I_{M,m}}}{Q_{M,m}(I_{M,m})} + \sum_{t=1}^{m-1}\xi_{M,I_{M,t}}\right\}.$$

*Then*

$$\widehat{R}_M = \frac{1}{M}\sum_{m=1}^{M}\gamma_{M,m}A_{M,m}.$$

*Proof.* We expand

$$\sum_{m=1}^{M}\gamma_{M,m}A_{M,m} = \sum_{m=1}^{M}\gamma_{M,m}\left[\frac{\xi_{M,I_{M,m}}}{NQ_{M,m}(I_{M,m})} + \frac{1}{N}\sum_{t=1}^{m-1}\xi_{M,I_{M,t}}\right].$$

Collect the coefficient of a fixed sampled value $\xi_{M,I_{M,t}}$, where $t \in \{1,\dots,M\}$. This coefficient is

$$\frac{\gamma_{M,t}}{NQ_{M,t}(I_{M,t})} + \sum_{m=t+1}^{M}\frac{\gamma_{M,m}}{N}.$$

The first term is given by

$$\frac{\gamma_{M,t}}{NQ_{M,t}(I_{M,t})} = \frac{N-M}{(N-t)(N-t+1)Q_{M,t}(I_{M,t})},$$

and for the second term we have

$$\sum_{m=t+1}^{M}\frac{\gamma_{M,m}}{N} = \sum_{m=t+1}^{M}\frac{N-M}{(N-m)(N-m+1)}.$$

Now, using

$$\frac{1}{k(k+1)} = \frac{1}{k} - \frac{1}{k+1},$$

and setting $k = N - m$, this telescopes to:

$$\sum_{m=t+1}^{M}\frac{N-M}{(N-m)(N-m+1)} = (N-M)\sum_{k=N-M}^{N-t-1}\frac{1}{k(k+1)} = 1 - \frac{N-M}{N-t}.$$

The coefficient of $\xi_{M,I_{M,t}}$ is therefore given by:

$$\begin{aligned}
&1 - \frac{N-M}{N-t} + \frac{N-M}{(N-t)(N-t+1)Q_{M,t}(I_{M,t})}. \\
&= 1 + \frac{N-M}{N-t}\left\{\frac{1}{(N-t+1)Q_{M,t}(I_{M,t})} - 1\right\} \\
&= V_{M,t}.
\end{aligned}$$

Finally, we have

$$\sum_{m=1}^{M} \gamma_{M,m} A_{M,m} = \sum_{t=1}^{M} V_{M,t}\xi_{M,I_{M,t}},$$

and dividing by $M$ gives the desired result. □

We now state a normalisation identity for the finite-pool weights.

**Lemma E.5** (Telescoping identity)**.** *For each $M$, we have $\sum_{m=1}^{M} \gamma_{M,m} = M$.*

*Proof.* Using the definition of $\gamma_{M,m}$,

$$\sum_{m=1}^{M} \gamma_{M,m} = N(N-M) \sum_{m=1}^{M} \frac{1}{(N-m)(N-m+1)}.$$

Setting $k = N - m$, we have:

$$\sum_{m=1}^{M} \frac{1}{(N-m)(N-m+1)} = \sum_{k=N-M}^{N-1} \frac{1}{k(k+1)}.$$

Now, using the identity

$$\frac{1}{k(k+1)} = \frac{1}{k} - \frac{1}{k+1},$$

the sum telescopes:

$$\sum_{k=N-M}^{N-1} \frac{1}{k(k+1)} = \frac{1}{N-M} - \frac{1}{N}.$$

Therefore

$$\sum_{m=1}^{M} \gamma_{M,m} = N(N-M)\left(\frac{1}{N-M} - \frac{1}{N}\right) = M.$$

□

### E.3 Finite–Sample Unbiasedness and Variance

Using the running-average representation of Lemma E.4, we now establish two finite–sample properties of our general formulation of LURE: that it is unbiased for the test–pool risk under any APP, and that its variance admits a clean decomposition into a sum of conditional variances of the importance–weighted increments. Using this, we then prove Prop. 3.1 from the main text

**Proposition E.6** (Unbiasedness and finite-sample variance)**.** *Assume that we have an APP in the sense of Def. E.1. Then, for each $m = 1, \ldots, M$, $\mathbb{E}[A_{M,m} \mid \mathcal{F}_{M,m-1}] = R_M$, and, consequently, $\mathbb{E}[A_{M,m}] = R_M$. Moreover, we have that*

$$\mathrm{Var}(A_{M,m}) = \frac{1}{N^2}\mathbb{E}\left[\mathrm{Var}\left(\frac{\xi_{M,I_{M,m}}}{Q_{M,m}(I_{M,m})} \;\middle|\; \mathcal{F}_{M,m-1}\right)\right],$$

*and for $m \neq k$, we have $\mathrm{Cov}(A_{M,m}, A_{M,k}) = 0$. It therefore follows that*

$$\mathbb{E}[\widehat{R}_M] = R_M$$

*and*

$$\mathrm{Var}(\widehat{R}_M) = \frac{1}{M^2}\sum_{m=1}^{M}\gamma_{M,m}^2\,\mathrm{Var}(A_{M,m}).$$

*Equivalently,*

$$\mathrm{Var}(\widehat{R}_M) = \frac{1}{M^2N^2}\sum_{m=1}^{M}\gamma_{M,m}^2\mathbb{E}\left[\mathrm{Var}\left(\frac{\xi_{M,I_{M,m}}}{Q_{M,m}(I_{M,m})}\;\middle|\;\mathcal{F}_{M,m-1}\right)\right].$$

*Proof.* Fix $m \in 1,\dots,M$. Conditional on $\mathcal{F}_{M,m-1}$, the set $S_{M,m}$, the proposal $Q_{M,m}$, and the previously sampled values $\xi_{M,I_{M,1}},\dots,\xi_{M,I_{M,m-1}}$ are fixed. Moreover, by admissibility we know that

$$\mathbb{P}(I_{M,m} = i \mid \mathcal{F}_{M,m-1}) = Q_{M,m}(i), \qquad i \in S_{M,m}.$$

Therefore, the conditional expectation of the importance-weighted sampled value is

$$\begin{aligned}\mathbb{E}\left[\frac{\xi_{M,I_{M,m}}}{NQ_{M,m}(I_{M,m})}\;\middle|\;\mathcal{F}_{M,m-1}\right] &= \sum_{i\in S_{M,m}} Q_{M,m}(i)\frac{\xi_{M,i}}{NQ_{M,m}(i)}\\ &= \frac{1}{N}\sum_{i\in S_{M,m}}\xi_{M,i}.\end{aligned}$$

Adding the contribution from the previously sampled indices gives

$$\begin{aligned}\mathbb{E}[A_{M,m} \mid \mathcal{F}_{M,m-1}] &= \frac{1}{N}\sum_{i\in S_{M,m}}\xi_{M,i} + \frac{1}{N}\sum_{t=1}^{m-1}\xi_{M,I_{M,t}}\\ &= \frac{1}{N}\sum_{i=1}^{N}\xi_{M,i}\\ &= R_M.\end{aligned}$$

Taking expectations then gives $\mathbb{E}[A_{M,m}] = R_M$. Now, noting that

$$A_{M,m} = \frac{1}{N}\frac{\xi_{M,I_{M,m}}}{Q_{M,m}(I_{M,m})} + \frac{1}{N}\sum_{t=1}^{m-1}\xi_{M,I_{M,t}},$$

where the second term is $\mathcal{F}_{M,m-1}$–measurable, we have

$$\mathrm{Var}(A_{M,m} \mid \mathcal{F}_{M,m-1}) = \frac{1}{N^2}\,\mathrm{Var}\left(\frac{\xi_{M,I_{M,m}}}{Q_{M,m}(I_{M,m})}\;\middle|\;\mathcal{F}_{M,m-1}\right).$$

Since $\mathbb{E}[A_{M,m} \mid \mathcal{F}_{M,m-1}] = R_M$ is deterministic, the law of total variance then yields the desired variance formula:

$$\begin{aligned}\mathrm{Var}(A_{M,m}) &= \mathbb{E}[\mathrm{Var}(A_{M,m} \mid \mathcal{F}_{M,m-1})]\\ &= \frac{1}{N^2}\mathbb{E}\left[\mathrm{Var}\left(\frac{\xi_{M,I_{M,m}}}{Q_{M,m}(I_{M,m})}\;\middle|\;\mathcal{F}_{M,m-1}\right)\right]\end{aligned}$$

Suppose now that $m < k$. Since $A_{M,m}$ is $\mathcal{F}_{M,k-1}$–measurable, we have

$$\begin{aligned}\mathrm{Cov}(A_{M,m}, A_{M,k}) &= \mathbb{E}[(A_{M,m} - R_M)(A_{M,k} - R_M)]\\ &= \mathbb{E}[(A_{M,m} - R_M)\mathbb{E}[A_{M,k} - R_M \mid \mathcal{F}_{M,k-1}]]\\ &= 0.\end{aligned}$$

The case $k < m$ follows by symmetry.

Finally, using Lemma E.4 and E.5 gives:

$$\begin{aligned}\mathbb{E}[\widehat{R}_M] &= \frac{1}{M}\sum_{m=1}^{M}\gamma_{M,m}\mathbb{E}[A_{M,m}] \\ &= \frac{R_M}{M}\sum_{m=1}^{M}\gamma_{M,m} \\ &= R_M.\end{aligned}$$

The variance formula follows similarly by using Lemma E.4 again and the fact that the $A_{M,m}$'s are pairwise uncorrelated. □

Prop. 3.1 follows as an immediate corollary to the above proposition. We restate the proposition more formally below.

**Corollary E.7** (Proposition 3.1)**.** *Consider the setup of §E.2 and fix any $\lambda \in \mathbb{R}$. Then, for any APP,*

$$\mathbb{E}\Big[\widehat{R}_M^{\mathrm{PPAT}}(\lambda)\Big] = R_M.$$

*Proof.* This follows immediately from Prop. E.6. Indeed, fix $\lambda \in \mathbb{R}$ and apply Prop. E.6 with the triangular array

$$\xi_{M,i} = \ell_i - \lambda c_{M,i} = \ell_i - \lambda(\widetilde{\ell}_i - \widetilde{R}_M).$$

The corresponding LURE estimator is exactly

$$\widehat{R}_M^{\mathrm{PPAT}}(\lambda) = \frac{1}{M}\sum_{m=1}^{M} V_{M,m}\xi_{M,I_{M,m}}.$$

Moreover, the finite-pool mean of this array is

$$\frac{1}{N}\sum_{i=1}^{N}\xi_{M,i} = \frac{1}{N}\sum_{i=1}^{N}\ell_i - \lambda\frac{1}{N}\sum_{i=1}^{N}(\widetilde{\ell}_i - \widetilde{R}_M) = R_M,$$

since $\frac{1}{N}\sum_{i=1}^{N}(\widetilde{\ell}_i - \widetilde{R}_M) = 0$. Prop. E.6 therefore gives

$$\mathbb{E}\Big[\widehat{R}_M^{\mathrm{PPAT}}(\lambda)\Big] = R_M,$$

as claimed. □

We now more formally state and prove Prop. 3.2 from the main text.

**Proposition E.8** (Proposition 3.2)**.** *Consider the setup of §E.2 and fix an APP $Q_{M,1:M}$. Let $c_{M,i} = \widetilde{\ell}_i - \widetilde{R}_M$ and define*

$$\widehat{R}_M^{\mathrm{LURE}} = \frac{1}{M}\sum_{m=1}^{M} V_{M,m}\,\ell_{I_{M,m}}, \qquad \widehat{C}_M^{\mathrm{LURE}} = \frac{1}{M}\sum_{m=1}^{M} V_{M,m}\,c_{M,I_{M,m}}.$$

*Then, for every $\lambda \in \mathbb{R}$,*

$$\mathrm{Var}\big(\widehat{R}_M^{\mathrm{PPAT}}(\lambda)\big) = \mathrm{Var}\big(\widehat{R}_M^{\mathrm{LURE}}\big) - 2\lambda\,\mathrm{Cov}\big(\widehat{R}_M^{\mathrm{LURE}}, \widehat{C}_M^{\mathrm{LURE}}\big) + \lambda^2\,\mathrm{Var}\big(\widehat{C}_M^{\mathrm{LURE}}\big).$$

*Thus, $\mathrm{Var}(\widehat{R}_M^{\mathrm{PPAT}}(\lambda)) < \mathrm{Var}(\widehat{R}_M^{\mathrm{LURE}})$ whenever $\lambda \in \big(\min\{0, 2\lambda^\star\}, \max\{0, 2\lambda^\star\}\big)$ and $\mathrm{Var}(\widehat{C}_M^{\mathrm{LURE}}) > 0$, where*

$$\lambda^\star = \frac{\mathrm{Cov}(\widehat{R}_M^{\mathrm{LURE}}, \widehat{C}_M^{\mathrm{LURE}})}{\mathrm{Var}(\widehat{C}_M^{\mathrm{LURE}})}.$$

*In particular, $\lambda^\star$ minimises the variance of $\widehat{R}_M^{\mathrm{PPAT}}(\lambda)$.*

*Proof.* Fix $\lambda \in \mathbb{R}$. By Remark E.3, $\widehat{R}_M^{\mathrm{PPAT}}(\lambda)$ is the LURE estimator associated with the residualised array $\zeta_{M,i}(\lambda) = \ell_i - \lambda c_{M,i}$. That is,

$$\begin{aligned}\widehat{R}_M^{\mathrm{PPAT}}(\lambda) &= \frac{1}{M}\sum_{m=1}^{M} V_{M,m}\,\zeta_{M,I_{M,m}}(\lambda) \\ &= \frac{1}{M}\sum_{m=1}^{M} V_{M,m}\big(\ell_{I_{M,m}} - \lambda c_{M,I_{M,m}}\big) \\ &= \widehat{R}_M^{\mathrm{LURE}} - \lambda\,\widehat{C}_M^{\mathrm{LURE}}.\end{aligned}$$

Now, by the properties of variance we have

$$\mathrm{Var}\big(\widehat{R}_M^{\mathrm{PPAT}}(\lambda)\big) = \mathrm{Var}\big(\widehat{R}_M^{\mathrm{LURE}}\big) - 2\lambda\,\mathrm{Cov}\big(\widehat{R}_M^{\mathrm{LURE}}, \widehat{C}_M^{\mathrm{LURE}}\big) + \lambda^2\,\mathrm{Var}\big(\widehat{C}_M^{\mathrm{LURE}}\big),$$

as claimed.

For the second part, write $V_C := \mathrm{Var}(\widehat{C}_M^{\mathrm{LURE}}) > 0$ and $b := \mathrm{Cov}(\widehat{R}_M^{\mathrm{LURE}}, \widehat{C}_M^{\mathrm{LURE}})$, and set

$$\phi(\lambda) := \mathrm{Var}\big(\widehat{R}_M^{\mathrm{PPAT}}(\lambda)\big) - \mathrm{Var}\big(\widehat{R}_M^{\mathrm{LURE}}\big) = V_C\,\lambda^2 - 2b\,\lambda = V_C\,\lambda\big(\lambda - 2\lambda^\star\big),$$

where $\lambda^\star = b/V_C$. This is a strictly convex parabola with roots $0$ and $2\lambda^\star$, so $\phi(\lambda) < 0$ if and only if $\lambda$ lies strictly between these roots, i.e. $\lambda \in (\min\{0, 2\lambda^\star\}, \max\{0, 2\lambda^\star\})$; this is the stated variance–reduction condition. Examining the derivatives of $\phi$ then gives $\lambda^\star$ as the minimiser of $\mathrm{Var}\big(\widehat{R}_M^{\mathrm{PPAT}}(\lambda)\big)$, as required. □

### E.3.1 PPAT Variance Bound

Here, we formally state and prove the variance bound of Prop. 3.4. Throughout we fix the budget $M$ and the finite test pool, and write $N$ for $N_M$. Recall from Remark E.3 that, for a fixed $\lambda \in \mathbb{R}$, the PPAT estimator is exactly the LURE estimator associated with the triangular array $\zeta_{M,i}(\lambda) = \ell_i - \lambda\, c_{M,i}$. That is, by the running–average representation of Lemma E.4,

$$\widehat{R}_M^{\mathrm{PPAT}}(\lambda) \;=\; \frac{1}{M}\sum_{m=1}^{M} V_{M,m}\,\zeta_{M,I_{M,m}}(\lambda) \;=\; \frac{1}{M}\sum_{m=1}^{M} \gamma_{M,m}\,A_{M,m},$$

where $\gamma_{M,m}$ and $A_{M,m}$ are as defined there for the array $\zeta_{M,\cdot}(\lambda)$.

**Proposition E.9** (Variance bound for PPAT)**.** *Suppose we are given an APP satisfying a uniform overlap condition: there exists $\beta > 0$ such that*

$$Q_{M,m}(i) \;\geq\; \frac{\beta}{N}, \qquad m = 1, \ldots, M,\;\; i \in S_{M,m}.$$

*Then, for every $\lambda \in \mathbb{R}$,*

$$\mathrm{Var}\big(\widehat{R}_M^{\mathrm{PPAT}}(\lambda)\big) \;\leq\; \frac{1}{\beta M}\,\frac{N}{N-M+1}\left(\frac{1}{N}\sum_{i=1}^{N}(\ell_i - \lambda c_{M,i})^2\right).$$

*Whenever $\frac{1}{N}\sum_{i=1}^{N} c_{M,i}^2 > 0$, the right–hand side is minimised over $\lambda$ by*

$$\lambda^\dagger \;=\; \frac{N^{-1}\sum_{i=1}^{N}\ell_i c_{M,i}}{N^{-1}\sum_{i=1}^{N} c_{M,i}^2}.$$

*Proof.* Fix $\lambda \in \mathbb{R}$, abbreviate $\zeta_i := \zeta_{M,i}(\lambda) = \ell_i - \lambda c_{M,i}$, and set $D(\lambda) := \frac{1}{N}\sum_{i=1}^{N}\zeta_i^2 = \frac{1}{N}\sum_{i=1}^{N}(\ell_i - \lambda c_{M,i})^2$.

By Prop. E.6, the running averages $A_{M,1}, \dots, A_{M,M}$ are pairwise uncorrelated and satisfy

$$\mathbb{E}[A_{M,m} \mid \mathcal{F}_{M,m-1}] = R_M,$$
$$\mathrm{Var}\big(\widehat{R}_M^{\mathrm{PPAT}}(\lambda)\big) = \frac{1}{M^2} \sum_{m=1}^{M} \gamma_{M,m}^2 \, \mathrm{Var}(A_{M,m}),$$
$$\mathrm{Var}(A_{M,m}) = \frac{1}{N^2} \, \mathbb{E}\left[\mathrm{Var}\left( \frac{\zeta_{I_{M,m}}}{Q_{M,m}(I_{M,m})} \,\middle|\, \mathcal{F}_{M,m-1} \right)\right].$$

We first bound $\mathrm{Var}(A_{M,m})$ uniformly in $m$. For this, fix an arbitrary $m$ and condition on $\mathcal{F}_{M,m-1}$. Bounding the conditional variance by the conditional second moment and evaluating the latter gives

$$\begin{aligned}
\mathrm{Var}\left( \frac{\zeta_{I_{M,m}}}{Q_{M,m}(I_{M,m})} \,\middle|\, \mathcal{F}_{M,m-1} \right) &\le \mathbb{E}\left[ \frac{\zeta_{I_{M,m}}^2}{Q_{M,m}(I_{M,m})^2} \,\middle|\, \mathcal{F}_{M,m-1} \right] \\
&= \sum_{i \in S_{M,m}} Q_{M,m}(i) \, \frac{\zeta_i^2}{Q_{M,m}(i)^2} \\
&= \sum_{i \in S_{M,m}} \frac{\zeta_i^2}{Q_{M,m}(i)}.
\end{aligned}$$

Now, the overlap condition implies $1/Q_{M,m}(i) \le N/\beta$ for every $i \in S_{M,m}$. As the summands are nonnegative and $S_{M,m} \subseteq \{1, \dots, N\}$,

$$\sum_{i \in S_{M,m}} \frac{\zeta_i^2}{Q_{M,m}(i)} \;\le\; \frac{N}{\beta} \sum_{i \in S_{M,m}} \zeta_i^2 \;\le\; \frac{N}{\beta} \sum_{i=1}^{N} \zeta_i^2.$$

The right–hand side is deterministic, so taking expectations and dividing by $N^2$ yields

$$\mathrm{Var}(A_{M,m}) \;\le\; \frac{1}{N^2} \cdot \frac{N}{\beta} \sum_{i=1}^{N} \zeta_i^2 \;=\; \frac{1}{\beta} \left( \frac{1}{N} \sum_{i=1}^{N} \zeta_i^2 \right) \;=\; \frac{D(\lambda)}{\beta},$$

a bound independent of $m$.

We now aggregate over the different rounds. The weights $\gamma_{M,m} = \frac{N(N-M)}{(N-m)(N-m+1)}$ are positive, and since $(N-m)(N-m+1)$ is decreasing in $m$ on $\{1, \dots, M\}$, $\gamma_{M,m}$ is increasing in $m$; its maximum is therefore attained at $m = M$,

$$\max_{1 \le m \le M} \gamma_{M,m} = \gamma_{M,M} = \frac{N(N-M)}{(N-M)(N-M+1)} = \frac{N}{N-M+1}.$$

Using $\gamma_{M,m}^2 \le \gamma_{M,m} \cdot \max_{1 \le k \le M} \gamma_{M,k}$ together with the identity $\sum_{m=1}^{M} \gamma_{M,m} = M$ of Lemma E.5, gives

$$\sum_{m=1}^{M} \gamma_{M,m}^2 \;\le\; \Big( \max_{1 \le k \le M} \gamma_{M,k} \Big) \sum_{m=1}^{M} \gamma_{M,m} \;=\; \frac{N}{N-M+1} \, M.$$

Combining this with the variance decomposition above then gives

$$\begin{aligned}
\mathrm{Var}\big(\widehat{R}_M^{\mathrm{PPAT}}(\lambda)\big) = \frac{1}{M^2} \sum_{m=1}^{M} \gamma_{M,m}^2 \, \mathrm{Var}(A_{M,m}) \;&\le\; \frac{1}{M^2} \cdot \frac{N}{N-M+1} \, M \cdot \frac{D(\lambda)}{\beta} \\
&= \frac{1}{\beta M} \, \frac{N}{N-M+1} \, D(\lambda),
\end{aligned}$$

as claimed.

Finally, only $D(\lambda)$ depends on $\lambda$. Expanding $D(\lambda)$ gives

$$D(\lambda) = \frac{1}{N}\sum_{i=1}^{N} \ell_i^2 - 2\lambda \frac{1}{N}\sum_{i=1}^{N} \ell_i c_{M,i} + \lambda^2 \frac{1}{N}\sum_{i=1}^{N} c_{M,i}^2,$$

a quadratic in $\lambda$ with leading coefficient $\frac{1}{N}\sum_i c_{M,i}^2$. When $\frac{1}{N}\sum_i c_{M,i}^2 > 0$ it is strictly convex, and its derivative vanishes at the unique point

$$\lambda^\dagger = \frac{N^{-1}\sum_{i=1}^{N} \ell_i c_{M,i}}{N^{-1}\sum_{i=1}^{N} c_{M,i}^2}.$$

As the constant multiplying $D(\lambda)$ is independent of $\lambda$, $\lambda^\dagger$ minimises the upper bound. □

Note that $\lambda = 0$ gives the LURE bound $\mathrm{Var}(\widehat{R}_M^{\mathrm{LURE}}) \leq \frac{\sum_{i=1}^{N} \ell_i^2}{\beta M(N-M+1)}$. PPAT at $\lambda^\dagger$ reduces this bound by $\frac{(\sum_{i=1}^{N} \ell_i c_i)^2}{\beta M(N-M+1)\sum_{i=1}^{N} c_i^2} \geq 0$, with strict improvement when $\sum_i \ell_i c_i \neq 0$.

### E.3.2 Tightness of the PPAT Variance Bound

Here, we discuss when the bound in Prop. E.9 is tight and, consequently, when we can expect $\lambda^\dagger$ to serve as a suitable alternative to $\lambda^\star$. In particular, we show that the tightness of the bound depends critically on how small $M/N$ is and how close the proposal is to being uniform.

We begin by noting that the bound is derived by chaining four separate inequalities to relax the exact variance. By tracking when each of these relaxations holds as an equality, we can better understand its overall tightness. As before, write $\zeta_i := \zeta_{M,i}(\lambda) = \ell_i - \lambda c_{M,i}$ for the residualised loss, $R_M = N^{-1}\sum_i \zeta_i$ for its pool mean, and $\sigma_\zeta^2 := N^{-1}\sum_i (\zeta_i - R_M)^2$ for its pool variance, so that $D(\lambda) = \sigma_\zeta^2 + R_M^2$. The proof passes through the following four relaxations:

(i) *Second-moment relaxation.* We bound the conditional variance of the round-$m$ increment by its conditional second moment, which discards the (nonnegative) squared mean of the *remaining* residuals, $N^{-2}\big(\sum_{i\in S_{M,m}} \zeta_i\big)^2$.

(ii) *Overlap relaxation.* We replace $1/Q_{M,m}(i)$ by its worst-case value $N/\beta$ based on the uniform overlap condition. This is exact only where the proposal sits exactly at the overlap floor, $Q_{M,m}(i) = \beta/N$, for every point with $\zeta_i \neq 0$. The more concentrated (non-uniform) the proposal is, the looser this step becomes.

(iii) *Pool-extension relaxation.* We enlarge the sum over the remaining pool $S_{M,m}$ to a sum over the full pool $\{1, \ldots, N\}$, discarding the squared residuals of the $m-1$ points already queried. This discarded mass is a vanishing fraction of the total whenever the number of acquired labels is small relative to the pool, $M \ll N$.

(iv) *Round-weight aggregation.* Finally, we bound $\sum_m \gamma_{M,m}^2$ by $\big(\max_m \gamma_{M,m}\big)\sum_m \gamma_{M,m} = \big(\max_m \gamma_{M,m}\big)M$. Since $\gamma_{M,m}$ is *strictly increasing* in $m$, this step is strict for any $M \geq 2$: the later rounds contribute disproportionately more to the bound than to a hypothetical "equal-weight" aggregation. The size of this gap is controlled by the spread of the weights across rounds,

$$\frac{\gamma_{M,M}}{\gamma_{M,1}} = \frac{N(N-1)}{(N-M)(N-M+1)} \longrightarrow 1 \quad \text{as } M/N \to 0,$$

i.e., when only a small fraction of the pool is queried, the weights $\gamma_{M,1}, \ldots, \gamma_{M,M}$ are all $\approx 1$, and this relaxation becomes negligible.

It's not difficult to see that the above relaxations are governed by two separate sets of conditions. Firstly, the relaxations (ii)-(iv) are all controlled by the same two knobs – *how small $M/N$ is* and *how close the proposal is to uniform* – and become negligible together as $M/N \to 0$ under a near-uniform proposal. Secondly, relaxation (i) depends only on whether the residualised losses are "centred", i.e. on the ratio $R_M^2/\sigma_\zeta^2$ between the squared pool mean and the pool variance of $\zeta_i$.

In the settings that motivate our approach, we expect relaxation (i) to be negligible. Its cost is controlled solely by the ratio $R_M^2/\sigma_\zeta^2$, which is small precisely when the residualised losses are highly variable across the pool relative to their average magnitude, $\sigma_\zeta^2 \gg R_M^2$. This is exactly the high-variance regime in which active testing is beneficial. Thus, we expect our bound to be tight when the budget is a small fraction of the pool ($M \ll N$) and our proposal is close to uniform. In particular, we note that $\lambda^\star$ coincides with $\lambda^\dagger$ when our proposal is uniform.

### E.4 Myopic Oracle Proposals

Here, we consider choosing proposals that minimise the variance of our estimator. As minimising the full–horizon variance is intractable, we instead target a tractable *myopic* proxy. We state our result for the general triangular array $\xi_{M,i}$ considered above; Prop. 3.3 in the main text then follows immediately by taking $\xi_{M,i} = \ell_i - \lambda(\widetilde{\ell}_i - \widetilde{R})$.

Recall from §E.2-E.3 that, for a general triangular array $\xi_{M,i}$, the estimator $\widehat{R}_M = \frac{1}{M}\sum_{m=1}^{M} V_{M,m}\,\xi_{M,I_{M,m}}$ admits the running–average representation of Lemma E.4 and the variance decomposition

$$\mathrm{Var}(\widehat{R}_M) = \frac{1}{M^2}\sum_{m=1}^{M} \gamma_{M,m}^2\ \mathrm{Var}(A_{M,m})$$

of Prop. E.6. Since the sum $\sum_{t=1}^{m-1}\xi_{M,I_{M,t}}$ in $A_{M,m}$ is $\mathcal{F}_{M,m-1}$–measurable and $\gamma_{M,m}$ is a deterministic constant, the only part of the round–$m$ contribution that depends on the proposal $Q_{M,m}$ is the conditional variance of the importance–weighted increment,

$$\mathrm{Var}\big(A_{M,m} \mid \mathcal{F}_{M,m-1}\big) = \frac{1}{N_M^2}\ \mathrm{Var}\left(\frac{\xi_{M,I_{M,m}}}{Q_{M,m}(I_{M,m})}\,\middle|\,\mathcal{F}_{M,m-1}\right).$$

The full–horizon variance–minimising proposal is intractable, because the choice of $Q_{M,m}$ also shapes the laws of all later pools $S_{M,m+1},\dots$. Following §3.2, we therefore adopt a *myopic* objective: at each round $m$, conditionally on the history $\mathcal{F}_{M,m-1}$, we choose the proposal minimising this conditional variance. The next proposition solves this problem in closed form for an arbitrary array.

**Proposition E.10** (Myopic oracle proposal)**.** *Fix $M \geq 1$ and a round $m \in \{1,\dots,M\}$, and condition on $\mathcal{F}_{M,m-1}$, so that the remaining pool $S_{M,m}$ and the values $(\xi_{M,i})_{i\in S_{M,m}}$ are fixed. Assume $\sum_{j\in S_{M,m}}|\xi_{M,j}| > 0$. Then, among all proposals supported on $S_{M,m}$, the conditional variance*

$$\mathcal{V}(Q_{M,m}) := \mathrm{Var}\left(\frac{\xi_{M,I_{M,m}}}{N_M\,Q_{M,m}(I_{M,m})}\,\middle|\,\mathcal{F}_{M,m-1}\right)$$

*is minimised by*

$$Q^\star_{M,m}(i) = \frac{|\xi_{M,i}|}{\sum_{j\in S_{M,m}}|\xi_{M,j}|},$$

*with minimal value*

$$\mathcal{V}\big(Q^\star_{M,m}\big) = \frac{1}{N_M^2}\left[\Big(\sum_{i\in S_{M,m}}|\xi_{M,i}|\Big)^2 - \Big(\sum_{i\in S_{M,m}}\xi_{M,i}\Big)^2\right] \geq 0.$$

*Proof.* Condition throughout on $\mathcal{F}_{M,m-1}$. To lighten notation, let $S = S_{M,m}$, $\xi_i = \xi_{M,i}$, $Q = Q_{M,m}$, $I = I_{M,m}$ and $N = N_M$. Set $W := \xi_I/(N\,Q(I))$.

Firstly, note that the conditional mean of $W$ does not depend on $Q$. Indeed,

$$\mathbb{E}[W \mid \mathcal{F}_{M,m-1}] = \sum_{i\in S} Q(i)\,\frac{\xi_i}{N\,Q(i)} = \frac{1}{N}\sum_{i\in S}\xi_i.$$

Now, because $\mathcal{V}(Q) = \mathbb{E}[W^2 \mid \mathcal{F}_{M,m-1}] - \big(\mathbb{E}[W \mid \mathcal{F}_{M,m-1}]\big)^2$ and the second term is constant in $Q$, minimising $\mathcal{V}(Q)$ is equivalent to minimising the conditional second moment

$$\mathbb{E}[W^2 \mid \mathcal{F}_{M,m-1}] = \sum_{i\in S} Q(i)\,\frac{\xi_i^2}{N^2\,Q(i)^2} = \frac{1}{N^2}\sum_{i\in S}\frac{\xi_i^2}{Q(i)}.$$

Applying the Cauchy–Schwarz inequality with $a_i = |\xi_i|/\sqrt{Q(i)}$ and $b_i = \sqrt{Q(i)}$ then gives

$$\Big(\sum_{i\in S}|\xi_i|\Big)^2 = \Big(\sum_{i\in S} a_i b_i\Big)^2 \le \Big(\sum_{i\in S} a_i^2\Big)\Big(\sum_{i\in S} b_i^2\Big) = \Big(\sum_{i\in S}\frac{\xi_i^2}{Q(i)}\Big)\underbrace{\sum_{i\in S} Q(i)}_{=1}.$$

Hence $\sum_{i\in S}\xi_i^2/Q(i) \ge \big(\sum_{i\in S}|\xi_i|\big)^2$. Since $b_i > 0$, equality holds iff $a_i = c\,b_i$ for some constant $c$, i.e. $Q(i) = |\xi_i|/c$ for all $i \in S$. Noting that $\sum_{i\in S} Q(i) = 1$, $c$ is forced to be $c = \sum_{j\in S}|\xi_j|$, giving

$$Q^\star(i) = \frac{|\xi_i|}{\sum_{j\in S}|\xi_j|}, \qquad i \in S,$$

which is a valid distribution because $\sum_{j\in S}|\xi_j| > 0$.

Now, substituting $Q^\star$ into the second moment yields $\frac{1}{N^2}\big(\sum_{i\in S}|\xi_i|\big)^2$; subtracting the constant squared mean $\frac{1}{N^2}\big(\sum_{i\in S}\xi_i\big)^2$ gives the stated minimal value of $\mathcal{V}$, which is non-negative by the triangle inequality $\big|\sum_i \xi_i\big| \le \sum_i |\xi_i|$. □

Prop. 3.3 follows as an immediate corollary to Prop. E.10.

**Corollary E.11** (Proposition 3.3)**.** *Consider the setup of §E.2 and fix $\lambda \in \mathbb{R}$. Take the PPAT array*

$$\xi_{M,i} = \ell_i - \lambda\big(\widetilde{\ell}_i - \tilde{R}_M\big)$$

*of Remark E.3, where $\widetilde{\ell}_i = L(f(x_i), g(x_i))$ and $\widetilde{R}_M = \frac{1}{N_M}\sum_{i=1}^{N_M}\widetilde{\ell}_i$. Fix a round $m \in \{1,\dots,M\}$ and condition on $\mathcal{F}_{M,m-1}$ and assume $\sum_{j\in S_{M,m}}\big|\ell_j - \lambda(\widetilde{\ell}_j - \tilde{R}_M)\big| > 0$. Then, among all proposals supported on $S_{M,m}$, the myopic proxy*

$$\mathcal{V}(Q_{M,m}) := \mathrm{Var}\left(\frac{\ell_{I_{M,m}} - \lambda\big(\widetilde{\ell}_{I_{M,m}} - \widetilde{R}_M\big)}{N_M\, Q_{M,m}(I_{M,m})}\;\middle|\;\mathcal{F}_{M,m-1}\right)$$

*is minimised by*

$$Q^\star_{M,m,\lambda}(i) = \frac{\big|\ell_i - \lambda(\widetilde{\ell}_i - \widetilde{R}_M)\big|}{\sum_{j\in S_{M,m}}\big|\ell_j - \lambda(\widetilde{\ell}_j - \widetilde{R}_M)\big|},$$

*This is exactly the myopic oracle proposal of Proposition 3.3.*

*Proof.* Fix $\lambda \in \mathbb{R}$. This is an application of Prop. E.10 to the triangular array $\xi_{M,i} = \ell_i - \lambda(\tilde{\ell}_i - \tilde{R}_M)$ of Remark E.3. With this array, $\xi_{M,I_{M,m}} = \ell_{I_{M,m}} - \lambda(\tilde{\ell}_{I_{M,m}} - \tilde{R}_M)$, so the conditional variance $\mathcal{V}(Q_{M,m})$ of Prop. E.10 is precisely the residualised myopic proxy in the statement, and the condition $\sum_{j\in S_{M,m}}|\xi_{M,j}| > 0$ holds by assumption. Prop. E.10 therefore yields that, among all proposals supported on $S_{M,m}$, the minimiser is

$$Q^\star_{M,m}(i) = \frac{|\xi_{M,i}|}{\sum_{j\in S_{M,m}}|\xi_{M,j}|} = \frac{\big|\ell_i - \lambda(\tilde{\ell}_i - \tilde{R}_M)\big|}{\sum_{j\in S_{M,m}}\big|\ell_j - \lambda(\tilde{\ell}_j - \tilde{R}_M)\big|} =: Q^\star_{M,m,\lambda}(i),$$

which is the stated proposal. □

Note that setting $\lambda = 0$ in Corr. E.11 recovers the LURE oracle $Q^\star_m(i) \propto |\ell_i|$ of §2.1.

### E.5 Asymptotics

We now establish the asymptotic guarantees underlying the confidence intervals used in the main text. We first work with our general triangular-array version of the LURE estimator, proving consistency, asymptotic normality, and the validity of a plug-in variance estimate for constructing confidence intervals. We then specialise these results to PPAT with a fixed value of $\lambda$, before extending the argument to the plug-in choice $\widehat{\lambda}_M$. We conclude by discussing the main assumptions required for these asymptotic results.

### E.5.1 Consistency of the General LURE Estimator

Here we establish that, under a uniform overlap condition on our proposals and a mild second–moment–like condition on the finite–pool array, the general LURE estimator is $L^2$–consistent for the test–pool risk; the proof proceeds by bounding the conditional variance of each running average using the overlap condition and aggregating across rounds.

**Theorem E.12** ($L^2$-consistency)**.** *Assume that $N_M/M \to \alpha > 1$. Assume also that there exists $\beta > 0$ such that, for all sufficiently large $M$,*

$$Q_{M,m}(i) \geq \frac{\beta}{N_M}, \qquad m = 1, \ldots, M, \quad i \in S_{M,m}.$$

*Then, for every $m = 1, \ldots, M$,*

$$\mathrm{Var}(A_{M,m}) \leq \frac{1}{\beta N_M} \sum_{i=1}^{N_M} \xi_{M,i}^2 = \frac{1}{\beta} \left( \frac{1}{N_M} \sum_{i=1}^{N_M} \xi_{M,i}^2 \right).$$

*If, in addition $\frac{1}{N_M} \sum_{i=1}^{N_M} \xi_{M,i}^2 = O(1)$, then*

$$\mathbb{E}[(\widehat{R}_M - R_M)^2] \to 0.$$

*More generally, the same conclusion holds if $\frac{1}{N_M} \sum_{i=1}^{N_M} \xi_{M,i}^2 = o(M)$.*

*Proof.* To simplify our notation, we write $N = N_M$. From the proof of Prop. E.6, we have

$$\mathrm{Var}(A_{M,m}) = \mathbb{E}[\mathrm{Var}(A_{M,m} \mid \mathcal{F}_{M,m-1})].$$

Moreover,

$$\mathrm{Var}(A_{M,m} \mid \mathcal{F}_{M,m-1}) \leq \mathbb{E}\left[ \left( \frac{\xi_{M,I_{M,m}}}{N Q_{M,m}(I_{M,m})} \right)^2 \middle| \mathcal{F}_{M,m-1} \right].$$

This gives:

$$\begin{aligned} \mathrm{Var}(A_{M,m} \mid \mathcal{F}_{M,m-1}) &\leq \sum_{i \in S_{M,m}} Q_{M,m}(i) \frac{\xi_{M,i}^2}{N^2 Q_{M,m}(i)^2} \\ &= \frac{1}{N^2} \sum_{i \in S_{M,m}} \frac{\xi_{M,i}^2}{Q_{M,m}(i)}. \end{aligned}$$

Now, using the overlap condition we get $\frac{1}{Q_{M,m}(i)} \leq \frac{N}{\beta}$ and consequently:

$$\mathrm{Var}(A_{M,m} \mid \mathcal{F}_{M,m-1}) \leq \frac{1}{N^2} \frac{N}{\beta} \sum_{i \in S_{M,m}} \xi_{M,i}^2 \leq \frac{1}{\beta N} \sum_{i=1}^{N} \xi_{M,i}^2.$$

Taking expectations then gives

$$\mathrm{Var}(A_{M,m}) \leq \frac{1}{\beta N} \sum_{i=1}^{N} \xi_{M,i}^2.$$

Moreover, since $N_M/M \to \alpha > 1$, the sequence $\max_{1 \leq m \leq M} \gamma_{M,m}$ is bounded. Indeed,

$$\gamma_{M,m} = \frac{N(N-M)}{(N-m)(N-m+1)} \leq \frac{N}{N-M+1},$$

and

$$\frac{N}{N-M+1} \to \frac{\alpha}{\alpha-1} < \infty.$$

Finally, using Prop. E.6, there exists a constant $C_\gamma < \infty$ such that, for all sufficiently large $M$,

$$\begin{aligned}\mathbb{E}[(\widehat{R}_M - R_M)^2] &= \mathrm{Var}(\widehat{R}_M) \\ &= \frac{1}{M^2}\sum_{m=1}^{M} \gamma_{M,m}^2 \,\mathrm{Var}(A_{M,m}) \\ &\le \frac{1}{M^2}\sum_{m=1}^{M} C_\gamma^2 \frac{1}{\beta N}\sum_{i=1}^{N} \xi_{M,i}^2 \\ &= \frac{C_\gamma^2}{\beta M}\left(\frac{1}{N}\sum_{i=1}^{N}\xi_{M,i}^2\right).\end{aligned}$$

If $N^{-1}\sum_i \xi_{M,i}^2 = O(1)$, the right–hand side is $O(M^{-1})$, and hence tends to zero. More generally, it tends to zero whenever $N^{-1}\sum_i \xi_{M,i}^2 = o(M)$. □

### E.5.2 Asymptotic Normality and Confidence Intervals for the General LURE Estimator

Building on the martingale structure exposed in §E.3, we now establish a new central limit theorem for $\sqrt{M}(\widehat{R}_M - R_M)$ under the uniform overlap condition and a $(2+\delta)$-moment condition, and we show that the asymptotic variance can be consistently estimated from the queried labels alone, yielding the asymptotically valid confidence intervals reported in §4. Our result is obtained by applying a martingale central limit theorem for triangular arrays [12, Theorem 35.12].

**Theorem E.13** (Martingale CLT for LURE)**.** *Assume $N_M/M \to \alpha > 1$. Assume that our APP satisfies a uniform overlap condition:*

$$Q_{M,m}(i) \ge \frac{\beta}{N_M}, \qquad m = 1,\dots,M, \quad i \in S_{M,m},$$

*for all sufficiently large $M$. For $m = 1,\dots,M$, define $s_{M,m}^2 = \mathrm{Var}(A_{M,m} \mid \mathcal{F}_{M,m-1})$ and $\sigma_M^2 = \frac{1}{M}\sum_{m=1}^{M}\gamma_{M,m}^2 s_{M,m}^2$.*

*Assume that $\sigma_M^2 \to_p \sigma^2$ for some $0 < \sigma^2 < \infty$. Assume also that, for some $\delta > 0$,*

$$\frac{1}{N_M}\sum_{i=1}^{N_M} |\xi_{M,i}|^{2+\delta} = O(1).$$

*Then*

$$\sqrt{M}(\widehat{R}_M - R_M) \Rightarrow N(0,\sigma^2).$$

*Proof.* Firstly, define the martingale–difference array $\Delta_{M,m} = \frac{\gamma_{M,m}}{\sqrt{M}}(A_{M,m} - R_M)$. By Proposition E.6, we have

$$\mathbb{E}[A_{M,m} - R_M \mid \mathcal{F}_{M,m-1}] = 0,$$

and thus

$$\mathbb{E}[\Delta_{M,m} \mid \mathcal{F}_{M,m-1}] = 0.$$

Now, using Lemma E.4 and Lemma E.5, we get

$$\begin{aligned}\sqrt{M}(\widehat{R}_M - R_M) &= \sqrt{M}\left[\frac{1}{M}\sum_{m=1}^{M}\gamma_{M,m}A_{M,m} - R_M\right] \\ &= \frac{1}{\sqrt{M}}\sum_{m=1}^{M}\gamma_{M,m}(A_{M,m} - R_M) \\ &= \sum_{m=1}^{M}\Delta_{M,m}.\end{aligned}$$

Here, the predictable quadratic variation is given by

$$\sum_{m=1}^{M} \mathbb{E}[\Delta_{M,m}^2 \mid \mathcal{F}_{M,m-1}] = \frac{1}{M} \sum_{m=1}^{M} \gamma_{M,m}^2 s_{M,m}^2 = \sigma_M^2.$$

Now, by assumption, we have $\sigma_M^2 \to_p \sigma^2$. It remains to verify the conditional Lindeberg condition.

Let $p = 2 + \delta$. As in the proof of Thm. E.12, the sequence $\max_{1 \le m \le M} \gamma_{M,m}$ is bounded. Moreover,

$$A_{M,m} - R_M = \frac{\xi_{M,I_{M,m}}}{NQ_{M,m}(I_{M,m})} - \frac{1}{N} \sum_{i \in S_{M,m}} \xi_{M,i}.$$

Using $|a-b|^p \le 2^{p-1}(|a|^p + |b|^p)$, we have

$$\begin{aligned}\mathbb{E}[|A_{M,m} - R_M|^p \mid \mathcal{F}_{M,m-1}] \le 2^{p-1}\mathbb{E}\left[\left|\frac{\xi_{M,I_{M,m}}}{NQ_{M,m}(I_{M,m})}\right|^p \middle| \mathcal{F}_{M,m-1}\right] \\ + 2^{p-1}\left|\frac{1}{N}\sum_{i \in S_{M,m}} \xi_{M,i}\right|^p.\end{aligned}$$

For the first term, our overlap condition gives

$$\begin{aligned}\mathbb{E}\left[\left|\frac{\xi_{M,I_{M,m}}}{NQ_{M,m}(I_{M,m})}\right|^p \middle| \mathcal{F}_{M,m-1}\right] &= \frac{1}{N^p} \sum_{i \in S_{M,m}} \frac{|\xi_{M,i}|^p}{Q_{M,m}(i)^{p-1}} \\ &\le \frac{1}{N^p}\left(\frac{N}{\beta}\right)^{p-1} \sum_{i \in S_{M,m}} |\xi_{M,i}|^p \\ &\le \beta^{1-p}\left(\frac{1}{N}\sum_{i=1}^{N} |\xi_{M,i}|^p\right).\end{aligned}$$

For the second term, we have

$$\left|\frac{1}{N}\sum_{i \in S_{M,m}} \xi_{M,i}\right|^p \le \left(\frac{1}{N}\sum_{i=1}^{N} |\xi_{M,i}|\right)^p \le \frac{1}{N}\sum_{i=1}^{N} |\xi_{M,i}|^p.$$

We therefore have that there exists a constant $C_{p,\beta} < \infty$ such that

$$\mathbb{E}[|A_{M,m} - R_M|^p \mid \mathcal{F}_{M,m-1}] \le C_{p,\beta}\left(\frac{1}{N}\sum_{i=1}^{N} |\xi_{M,i}|^p\right).$$

Consequently, for another finite constant $C < \infty$, we have

$$\begin{aligned}\sum_{m=1}^{M} \mathbb{E}[|\Delta_{M,m}|^p \mid \mathcal{F}_{M,m-1}] &= \sum_{m=1}^{M} \frac{\gamma_{M,m}^p}{M^{p/2}} \mathbb{E}[|A_{M,m} - R_M|^p \mid \mathcal{F}_{M,m-1}] \\ &\le CM^{1-p/2}\left(\frac{1}{N}\sum_{i=1}^{N} |\xi_{M,i}|^p\right) \\ &= CM^{-\delta/2}\left(\frac{1}{N}\sum_{i=1}^{N} |\xi_{M,i}|^{2+\delta}\right) \\ &\to 0.\end{aligned}$$

Hence, for every $\varepsilon > 0$,

$$\sum_{m=1}^{M} \mathbb{E}[\Delta_{M,m}^2 \mathbf{1}\{|\Delta_{M,m}| > \varepsilon\} \mid \mathcal{F}_{M,m-1}] \le \varepsilon^{-\delta} \sum_{m=1}^{M} \mathbb{E}[|\Delta_{M,m}|^{2+\delta} \mid \mathcal{F}_{M,m-1}] \to 0.$$

This is the conditional Lindeberg condition.

We can now apply the martingale central limit theorem for triangular arrays: if a martingale–difference array has predictable quadratic variation converging in probability to $\sigma^2$ and satisfies the conditional Lindeberg condition, then its sum converges in distribution to $N(0, \sigma^2)$. Concretely, we have

$$\sum_{m=1}^{M} \Delta_{M,m} \Rightarrow N(0, \sigma^2).$$

Using $\sum_{m=1}^{M} \Delta_{M,m} = \sqrt{M}(\widehat{R}_M - R_M)$, then gives our desired result. □

**Remark E.14** (Explicit form of the predictable variance)**.** *The conditional variance $s_{M,m}^2$ can be written explicitly as*

$$s_{M,m}^2 = \mathrm{Var}\left( \frac{\xi_{M,I_{M,m}}}{N Q_{M,m}(I_{M,m})} \,\middle|\, \mathcal{F}_{M,m-1} \right).$$

*Equivalently,*

$$s_{M,m}^2 = \frac{1}{N^2} \left[ \sum_{i \in S_{M,m}} \frac{\xi_{M,i}^2}{Q_{M,m}(i)} - \left( \sum_{i \in S_{M,m}} \xi_{M,i} \right)^2 \right].$$

*Thus, the asymptotic variance in Thm. E.13 is the limit of*

$$\sigma_M^2 = \frac{1}{M} \sum_{m=1}^{M} \gamma_{M,m}^2 \frac{1}{N^2} \left[ \sum_{i \in S_{M,m}} \frac{\xi_{M,i}^2}{Q_{M,m}(i)} - \left( \sum_{i \in S_{M,m}} \xi_{M,i} \right)^2 \right].$$

**Theorem E.15** (Consistent estimation of the CLT variance and studentized CLT)**.** *Assume the conditions of Thm. E.13. In addition, assume $\frac{1}{N_M} \sum_{i=1}^{N_M} |\xi_{M,i}|^4 = O(1)$. Moreover, define*

$$\widehat{\sigma}_M^2 = \frac{1}{M} \sum_{m=1}^{M} \gamma_{M,m}^2 (A_{M,m} - \widehat{R}_M)^2.$$

*Then $\widehat{\sigma}_M^2 \to_p \sigma^2$. Consequently, we have*

$$\frac{\sqrt{M}(\widehat{R}_M - R_M)}{\widehat{\sigma}_M} \Rightarrow N(0,1).$$

*An approximate $(1-\eta)$–level confidence interval for $R_M$ is given by:*

$$\widehat{R}_M \pm \Phi^{-1}(1 - \eta/2) \frac{\widehat{\sigma}_M}{\sqrt{M}}, \tag{25}$$

*where $\Phi$ is the CDF of the standard normal distribution.*

*Proof.* Define the infeasible realised quadratic variation

$$\widetilde{\sigma}_M^2 := \frac{1}{M} \sum_{m=1}^{M} \gamma_{M,m}^2 (A_{M,m} - R_M)^2.$$

We first show that $\widetilde{\sigma}_M^2 - \sigma_M^2 \to_p 0$. Let

$$Y_{M,m} = \frac{\gamma_{M,m}^2}{M} \left[ (A_{M,m} - R_M)^2 - s_{M,m}^2 \right].$$

Then $\mathbb{E}[Y_{M,m} \mid \mathcal{F}_{M,m-1}] = 0$. Thus $\sum_{m=1}^{M} Y_{M,m}$ is a martingale. Since $\max_m \gamma_{M,m}$ is bounded and, by the same overlap argument used in Thm. E.13,

$$\mathbb{E}[|A_{M,m} - R_M|^4 \mid \mathcal{F}_{M,m-1}] \leq C \left( \frac{1}{N} \sum_{i=1}^{N} |\xi_{M,i}|^4 \right),$$

we have

$$\mathbb{E}\left[(\widetilde{\sigma}_M^2 - \sigma_M^2)^2\right] \leq \frac{C}{M} \left( \frac{1}{N} \sum_{i=1}^{N} |\xi_{M,i}|^4 \right) \to 0.$$

Consequently, $\widetilde{\sigma}_M^2 - \sigma_M^2 \to_p 0$. Now, since $\sigma_M^2 \to_p \sigma^2$, it therefore follows that $\widetilde{\sigma}_M^2 \to_p \sigma^2$.

It now remains to replace $R_M$ by $\widehat{R}_M$. Let $D_{M,m} = A_{M,m} - R_M$. Then $A_{M,m} - \widehat{R}_M = D_{M,m} - (\widehat{R}_M - R_M)$ and, consequently,

$$\widehat{\sigma}_M^2 - \widetilde{\sigma}_M^2 = -2(\widehat{R}_M - R_M) \frac{1}{M} \sum_{m=1}^{M} \gamma_{M,m}^2 D_{M,m} + (\widehat{R}_M - R_M)^2 \frac{1}{M} \sum_{m=1}^{M} \gamma_{M,m}^2.$$

By Thm. E.12[14], we have $\widehat{R}_M - R_M = o_p(1)$. Note that we also have:

$$\frac{1}{M} \sum_{m=1}^{M} \gamma_{M,m}^2 = O(1), \tag{26}$$

$$\frac{1}{M} \sum_{m=1}^{M} \gamma_{M,m}^2 D_{M,m} = O_p(1), \tag{27}$$

where (26) follows from the boundedness of $\gamma_{M,m}$, and (27) follows from $\widetilde{\sigma}_M^2 \to_p \sigma^2$ and an application of the Cauchy-Schwarz inequality:

$$\left| \frac{1}{M} \sum_{m=1}^{M} \gamma_{M,m}^2 D_{M,m} \right| \leq \left( \frac{1}{M} \sum_{m=1}^{M} \gamma_{M,m}^2 \right)^{1/2} \left( \frac{1}{M} \sum_{m=1}^{M} \gamma_{M,m}^2 D_{M,m}^2 \right)^{1/2}.$$

Consequently, we have $\widehat{\sigma}_M^2 - \widetilde{\sigma}_M^2 = o_p(1)$. Combining this with $\widetilde{\sigma}_M^2 \to_p \sigma^2$ then yields:

$$\widehat{\sigma}_M^2 \to_p \sigma^2.$$

The studentised CLT follows from Thm. E.13 and Slutsky's theorem. □

### E.5.3 Fixed–$\lambda$ PPAT

Here, we prove Thm. 4.1 for the fixed-$\lambda$ case; the plug-in case is treated in §E.5.4. This follows as an immediate corollary to the theorems in §E.5.1-E.5.2. The key observation is that, for any fixed $\lambda \in \mathbb{R}$, PPAT is exactly the general LURE estimator of §E.2 applied to the residualised triangular array

$$\zeta_{M,i}(\lambda) = \ell_i - \lambda\, c_{M,i}, \qquad c_{M,i} = \widetilde{\ell}_i - \widetilde{R}_M,$$

whose finite–pool mean is the test–pool risk $R_M$; the consistency, central–limit, and variance–estimation results then transfer directly once the moment conditions on this array are translated into separate conditions on the true and proxy losses.

Recall from Remark E.14 that, for the array $\zeta_{M,\cdot}(\lambda)$, the predictable quadratic variation $\sigma_M^2$ of Thm. E.13 takes the explicit form

$$\sigma_M^2(\lambda) = \frac{1}{M} \sum_{m=1}^{M} \gamma_{M,m}^2 \frac{1}{N_M^2} \left[ \sum_{i \in S_{M,m}} \frac{\zeta_{M,i}(\lambda)^2}{Q_{M,m}(i)} - \Big( \sum_{i \in S_{M,m}} \zeta_{M,i}(\lambda) \Big)^2 \right]. \tag{28}$$

[14] This applies here as the fourth moment condition implies the second moment condition.

**Corollary E.16** (Fixed–$\lambda$ PPAT, Theorem 4.1)**.** *Consider the setup of §E.2 and fix $\lambda \in \mathbb{R}$. Assume that the pool ratio satisfies $N_M/M \to \alpha > 1$ and that our APP satisfies a uniform overlap condition: there exists $\beta > 0$ such that, for all sufficiently large $M$,*

$$Q_{M,m}(i) \geq \frac{\beta}{N_M}, \qquad m = 1, \dots, M, \;\; i \in S_{M,m}.$$

*Then the following hold.*

1. (Consistency.) *Assume $\frac{1}{N_M}\sum_{i=1}^{N_M} \ell_i^2 = O(1)$ and $\frac{1}{N_M}\sum_{i=1}^{N_M} \widetilde{\ell}_i^2 = O(1)$, then*
$$\widehat{R}_M^{\mathrm{PPAT}}(\lambda) - R_M \xrightarrow{\;p\;} 0.$$

2. (Asymptotic normality and confidence intervals.) *Assume, additionally, $\frac{1}{N_M}\sum_{i=1}^{N_M} \ell_i^4 = O(1)$ and $\frac{1}{N_M}\sum_{i=1}^{N_M} \tilde{\ell}_i^4 = O(1)$, and the predictable quadratic variation (28) satisfies $\sigma_M^2(\lambda) \xrightarrow{p} \sigma_{\mathrm{PPAT}}^2(\lambda)$ for some $\sigma_{\mathrm{PPAT}}^2(\lambda) \in (0, \infty)$, then*
$$\sqrt{M}\left\{\widehat{R}_M^{\mathrm{PPAT}}(\lambda) - R_M\right\} \Rightarrow \mathcal{N}\big(0, \sigma_{\mathrm{PPAT}}^2(\lambda)\big).$$

   *Moreover, writing $A_{M,m}(\lambda)$ for the running average of Lemma E.4 applied to the array $\zeta_{M,\cdot}(\lambda)$, i.e.*
$$A_{M,m}(\lambda) = \frac{1}{N_M}\left(\frac{\zeta_{M,I_{M,m}}(\lambda)}{Q_{M,m}(I_{M,m})} + \sum_{t=1}^{m-1} \zeta_{M,I_{M,t}}(\lambda)\right),$$
   *the plug–in estimator*
$$\widehat{\sigma}_M^2(\lambda) = \frac{1}{M}\sum_{m=1}^{M} \gamma_{M,m}^2\big(A_{M,m}(\lambda) - \widehat{R}_M^{\mathrm{PPAT}}(\lambda)\big)^2$$
   *satisfies $\widehat{\sigma}_M^2(\lambda) \xrightarrow{p} \sigma_{\mathrm{PPAT}}^2(\lambda)$, and hence*
$$\frac{\sqrt{M}\left\{\widehat{R}_M^{\mathrm{PPAT}}(\lambda) - R_M\right\}}{\widehat{\sigma}_M(\lambda)} \Rightarrow \mathcal{N}(0,1).$$
   *In particular,*
$$\widehat{R}_M^{\mathrm{PPAT}}(\lambda) \;\pm\; \Phi^{-1}\big(1 - \tfrac{\eta}{2}\big)\,\frac{\widehat{\sigma}_M(\lambda)}{\sqrt{M}}$$
   *is an asymptotic $(1-\eta)$–level confidence interval for $R_M$, where $\Phi$ is the CDF for the standard normal distribution.*

*Proof.* By Remark E.3, for any fixed $\lambda \in \mathbb{R}$ the PPAT estimator $\widehat{R}_M^{\mathrm{PPAT}}(\lambda)$ is exactly the generic LURE estimator of §E.2 applied to the array $\xi_{M,i} = \zeta_{M,i}(\lambda) = \ell_i - \lambda\, c_{M,i}$. Since $c_{M,i} = \tilde{\ell}_i - \tilde{R}_M$ has zero pool average, $\frac{1}{N_M}\sum_{i=1}^{N_M} c_{M,i} = 0$, the finite–pool mean of this array is

$$\frac{1}{N_M}\sum_{i=1}^{N_M} \zeta_{M,i}(\lambda) = \frac{1}{N_M}\sum_{i=1}^{N_M} \ell_i - \lambda\,\frac{1}{N_M}\sum_{i=1}^{N_M} c_{M,i} = R_M.$$

Hence the quantity "$R_M$" appearing in Thms. E.12, E.13 and E.15 for this array is precisely the test–pool risk, and it only remains to translate the moment conditions on $\xi_{M,i}$ into conditions on $\ell_i$ and $\tilde{\ell}_i$.

It remains to translate the moment assumptions on $\xi_{M,i}$ into the corresponding assumptions on $\ell_i$ and $\tilde{\ell}_i$. Fix $p \geq 1$. Since $c_{M,i} = \tilde{\ell}_i - \tilde{R}_M$ and $\tilde{R}_M$ is the pool average of the $\tilde{\ell}_i$, Jensen's inequality gives $|\tilde{R}_M|^p \leq \frac{1}{N_M}\sum_j |\tilde{\ell}_j|^p$, so that, using $|a-b|^p \leq 2^{p-1}(|a|^p + |b|^p)$,

$$\frac{1}{N_M}\sum_{i=1}^{N_M} |c_{M,i}|^p \leq 2^{p-1}\left(\frac{1}{N_M}\sum_{i=1}^{N_M} |\tilde{\ell}_i|^p + |\tilde{R}_M|^p\right) \leq C_p\,\frac{1}{N_M}\sum_{i=1}^{N_M} |\tilde{\ell}_i|^p,$$

for a constant $C_p < \infty$. As $\lambda$ is fixed, applying the same inequality to $\zeta_{M,i}(\lambda) = \ell_i - \lambda c_{M,i}$ yields

$$\frac{1}{N_M}\sum_{i=1}^{N_M}|\zeta_{M,i}(\lambda)|^p \le C_{p,\lambda}\left(\frac{1}{N_M}\sum_{i=1}^{N_M}|\ell_i|^p + \frac{1}{N_M}\sum_{i=1}^{N_M}|\tilde{\ell}_i|^p\right), \tag{29}$$

for a constant $C_{p,\lambda} < \infty$.

We first establish consistency. Taking $p = 2$ in (29), the assumptions $\frac{1}{N_M}\sum \ell_i^2 = O(1)$ and $\frac{1}{N_M}\sum \tilde{\ell}_i^2 = O(1)$ imply $\frac{1}{N_M}\sum \xi_{M,i}^2 = O(1)$. Thm. E.12 then gives $\mathbb{E}[(\widehat{R}_M^{\mathrm{PPAT}}(\lambda) - R_M)^2] \to 0$, and $L^2$ convergence implies $\widehat{R}_M^{\mathrm{PPAT}}(\lambda) - R_M \xrightarrow{p} 0$.

We now turn to the central limit theorem. Taking $p = 4$ in (29), the fourth–moment assumptions give $\frac{1}{N_M}\sum |\xi_{M,i}|^4 = O(1)$, and in particular the $(2+\delta)$–moment condition $\frac{1}{N_M}\sum |\xi_{M,i}|^{2+\delta} = O(1)$ with $\delta = 2$. The assumed stabilisation $\sigma_M^2(\lambda) \xrightarrow{p} \sigma_{\mathrm{PPAT}}^2(\lambda) \in (0,\infty)$ is exactly the predictable–quadratic–variation condition of Thm. E.13 for the array $\zeta_{M,\cdot}(\lambda)$. Thm. E.13 therefore yields

$$\sqrt{M}\big\{\widehat{R}_M^{\mathrm{PPAT}}(\lambda) - R_M\big\} \;\Rightarrow\; \mathcal{N}\big(0, \sigma_{\mathrm{PPAT}}^2(\lambda)\big).$$

Finally, we address estimation of the asymptotic variance and the asymptotic confidence interval. The bound $\frac{1}{N_M}\sum |\xi_{M,i}|^4 = O(1)$ established above is precisely the additional condition required by Thm. E.15. Applying that theorem to the array $\zeta_{M,\cdot}(\lambda)$ shows that the plug–in estimator is consistent, $\widehat{\sigma}_M^2(\lambda) \xrightarrow{p} \sigma_{\mathrm{PPAT}}^2(\lambda)$, and, by Slutsky's theorem, gives both the studentised limit and the asymptotic $(1-\eta)$–level confidence interval for $R_M$. □

**Remark E.17.** *Setting $\lambda = 0$ gives $\zeta_{M,i}(0) = \ell_i$ and $\widehat{R}_M^{\mathrm{PPAT}}(0) = \widehat{R}_M^{\mathrm{LURE}}$, so Corr. E.16 specialises to the corresponding consistency, asymptotic normality, and confidence–interval statements for LURE.*

### E.5.4 Plug–in PPAT

Finally, we extend the results in §E.5.3 for the plug–in PPAT estimator, where the tuning parameter $\lambda$ is replaced by the data-driven estimate $\widehat{\lambda}_M$ in §3.3. The proof compares the plug–in estimator with the PPAT estimator that uses the finite-pool oracle coefficient $\lambda_M^\dagger$. We show that replacing $\lambda_M^\dagger$ by $\widehat{\lambda}_M$ changes the estimator only by $o_p(M^{-1/2})$. Thus, at the $\sqrt{M}$ rate, the plug–in estimator has the same asymptotic behaviour as this oracle estimator: it is consistent, satisfies the same central limit theorem, and yields asymptotically valid confidence intervals when the variance is estimated from the data. We additionally quantify the bias introduced in our estimator when using $\widehat{\lambda}_M$ and show that this vanishes at a rate of $O(M^{-1})$.

To simplify our notation, for any given triangular array $a_{M,1:N_M}$ we will write

$$\bar{a}_M = \frac{1}{N_M}\sum_{i=1}^{N_M} a_{M,i}, \qquad \widehat{\mu}_M(a) = \frac{1}{M}\sum_{m=1}^{M} V_{M,m}\, a_{M,I_{M,m}}.$$

In this way, $\widehat{\mu}_M(a)$ is the LURE estimator applied to the array $a$. As before, define

$$c_{M,i} = \widetilde{\ell}_i - \widetilde{R}_M, \qquad \widetilde{R}_M = \frac{1}{N_M}\sum_{i=1}^{N_M}\widetilde{\ell}_i,$$

and note that $\bar{c}_M = 0$. Moreover, let

$$D_M = \frac{1}{N_M}\sum_{i=1}^{N_M} c_{M,i}^2, \qquad G_M = \frac{1}{N_M}\sum_{i=1}^{N_M}\ell_i c_{M,i},$$

such that the oracle finite-pool coefficient in §3.3 can be written as $\lambda_M^\dagger = \frac{G_M}{D_M}$, whenever $D_M > 0$.

As discussed in §3.3, $c_{M,i}$ is known for all test points so $D_M$ can be computed exactly. On the other hand, we estimate $G_M$ via LURE,

$$\widehat{G}_M = \widehat{\mu}_M(\ell c) = \frac{1}{M}\sum_{m=1}^{M} V_{M,m}\ell_{I_{M,m}}c_{M,I_{M,m}},$$

and set $\widehat{\lambda}_M = \frac{\widehat{G}_M}{D_M}$. Using the notation defined earlier, the plug-in PPAT estimator is

$$\widehat{R}_M^{\mathrm{PPAT}}(\widehat{\lambda}_M) = \widehat{\mu}_M(\ell) - \widehat{\lambda}_M\widehat{\mu}_M(c).$$

We now formally state and prove the consistency and asymptotic normality of the plug-in PPAT estimator. We prove the confidence interval result as a corollary in Corr. E.19.

**Theorem E.18** (Consistency and asymptotic normality of plug-in PPAT, Theorem 4.1)**.** *Assume that the pool ratio satisfies $N_M/M \to \alpha > 1$ and that the APP satisfies a uniform overlap condition: there exists $\beta > 0$ such that, for all sufficiently large $M$,*

$$Q_{M,m}(i) \geq \frac{\beta}{N_M}, \qquad m = 1, \ldots, M, \quad i \in S_{M,m}.$$

*Assume also that there exists $d > 0$ such that, for all sufficiently large $M$,*

$$D_M = \frac{1}{N_M}\sum_{i=1}^{N_M} c_{M,i}^2 \geq d.$$

*Then the following hold.*

1. (Consistency.) *Assume*

$$\frac{1}{N_M}\sum_{i=1}^{N_M}\ell_i^2 = O(1), \qquad \frac{1}{N_M}\sum_{i=1}^{N_M}c_{M,i}^2 = O(1), \qquad \frac{1}{N_M}\sum_{i=1}^{N_M}\ell_i^2 c_{M,i}^2 = O(1).$$

   *Then*

$$\widehat{\lambda}_M - \lambda_M^\dagger = O_p(M^{-1/2}) \qquad \text{and} \qquad \widehat{R}_M^{\mathrm{PPAT}}(\widehat{\lambda}_M) - R_M = O_p(M^{-1/2}).$$

   *In particular, both quantities are $o_p(1)$, so $\widehat{R}_M^{\mathrm{PPAT}}(\widehat{\lambda}_M) - R_M \to_p 0$ and $\widehat{\lambda}_M - \lambda_M^\dagger \to_p 0$.*

2. (Asymptotic normality.) *Assume, in addition, that for some $\delta > 0$,*

$$\frac{1}{N_M}\sum_{i=1}^{N_M}|\ell_i|^{2+\delta} = O(1), \qquad \frac{1}{N_M}\sum_{i=1}^{N_M}|c_{M,i}|^{2+\delta} = O(1),$$

   *and define the oracle residualised array $z_{M,i}^\dagger = \ell_i - \lambda_M^\dagger c_{M,i}$, with running averages and predictable variances*

$$A_{M,m}(z^\dagger) = \frac{1}{N_M}\left\{\frac{z_{M,I_{M,m}}^\dagger}{Q_{M,m}(I_{M,m})} + \sum_{t=1}^{m-1} z_{M,I_{M,t}}^\dagger\right\},$$

$$s_{M,m}^2(z^\dagger) = \mathrm{Var}\big(A_{M,m}(z^\dagger) \mid \mathcal{F}_{M,m-1}\big).$$

   *If the predictable quadratic variation stabilises,*

$$\sigma_{M,\dagger}^2 := \frac{1}{M}\sum_{m=1}^{M}\gamma_{M,m}^2 s_{M,m}^2(z^\dagger) \to_p \sigma_\dagger^2$$

   *for some $0 < \sigma_\dagger^2 < \infty$, then*

$$\sqrt{M}\left\{\widehat{R}_M^{\mathrm{PPAT}}(\widehat{\lambda}_M) - R_M\right\} \Rightarrow N(0, \sigma_\dagger^2).$$

*Proof.* Throughout we apply the $L^2$-consistency result (Thm. E.12) and the martingale central limit theorem for LURE (Thm. E.13) to fixed finite-pool arrays.

We first bound $\widehat{\lambda}_M - \lambda_M^\dagger$. Since

$$\widehat{\lambda}_M - \lambda_M^\dagger = \frac{\widehat{G}_M - G_M}{D_M}$$

and $D_M \geq d > 0$, it suffices to control $\widehat{G}_M - G_M$. Here $\widehat{G}_M = \widehat{\mu}_M(\ell c)$ is the LURE estimator of the finite-pool mean $G_M$ of the array $g_{M,i} := \ell_i c_{M,i}$, which satisfies

$$\frac{1}{N_M}\sum_{i=1}^{N_M} g_{M,i}^2 = \frac{1}{N_M}\sum_{i=1}^{N_M} \ell_i^2 c_{M,i}^2 = O(1).$$

Thm. E.12 applied to $g_{M,i}$ therefore gives $\widehat{G}_M - G_M = O_p(M^{-1/2})$, and hence

$$\widehat{\lambda}_M - \lambda_M^\dagger = O_p(M^{-1/2}).$$

Similarly, the centred proxy array $c_{M,i}$ has zero finite-pool mean, $\bar{c}_M = 0$, and satisfies $\frac{1}{N_M}\sum_i c_{M,i}^2 = O(1)$, so Thm. E.12 applied to $c_{M,i}$ gives

$$\widehat{\mu}_M(c) = \widehat{\mu}_M(c) - \bar{c}_M = O_p(M^{-1/2}).$$

We next describe the properties of the oracle residualised array $z_{M,i}^\dagger = \ell_i - \lambda_M^\dagger c_{M,i}$ needed for our results. First, $\lambda_M^\dagger$ is bounded: by Cauchy–Schwarz,

$$|G_M| = \left|\frac{1}{N_M}\sum_{i=1}^{N_M} \ell_i c_{M,i}\right| \leq \left(\frac{1}{N_M}\sum_{i=1}^{N_M} \ell_i^2\right)^{1/2} \left(\frac{1}{N_M}\sum_{i=1}^{N_M} c_{M,i}^2\right)^{1/2},$$

so that

$$|\lambda_M^\dagger| = \frac{|G_M|}{D_M} \leq \left(\frac{N_M^{-1}\sum_i \ell_i^2}{D_M}\right)^{1/2} = O(1),$$

using $D_M \geq d > 0$ and $N_M^{-1}\sum_i \ell_i^2 = O(1)$. Second, since $\bar{c}_M = 0$, the array $z^\dagger$ has finite-pool mean

$$\frac{1}{N_M}\sum_{i=1}^{N_M} z_{M,i}^\dagger = \frac{1}{N_M}\sum_{i=1}^{N_M} \ell_i - \lambda_M^\dagger \frac{1}{N_M}\sum_{i=1}^{N_M} c_{M,i} = R_M,$$

and, using $(a-b)^2 \leq 2a^2 + 2b^2$ together with the boundedness of $\lambda_M^\dagger$,

$$\frac{1}{N_M}\sum_{i=1}^{N_M} (z_{M,i}^\dagger)^2 \leq 2\frac{1}{N_M}\sum_{i=1}^{N_M} \ell_i^2 + 2(\lambda_M^\dagger)^2 \frac{1}{N_M}\sum_{i=1}^{N_M} c_{M,i}^2 = O(1).$$

Thm. E.12 applied to the fixed array $z^\dagger$ therefore gives

$$\widehat{\mu}_M(z^\dagger) - R_M = O_p(M^{-1/2}).$$

We now prove consistency. Writing $\widehat{R}_M^{\mathrm{PPAT}}(\widehat{\lambda}_M) = \widehat{\mu}_M(\ell) - \widehat{\lambda}_M \widehat{\mu}_M(c)$ and using the linearity of $\widehat{\mu}_M$,

$$\begin{aligned}
\widehat{R}_M^{\mathrm{PPAT}}(\widehat{\lambda}_M) - R_M &= \widehat{\mu}_M(\ell) - \widehat{\lambda}_M \widehat{\mu}_M(c) - R_M \\
&= \widehat{\mu}_M(\ell) - \lambda_M^\dagger \widehat{\mu}_M(c) - R_M - (\widehat{\lambda}_M - \lambda_M^\dagger)\widehat{\mu}_M(c) \\
&= \big(\widehat{\mu}_M(z^\dagger) - R_M\big) - (\widehat{\lambda}_M - \lambda_M^\dagger)\widehat{\mu}_M(c).
\end{aligned}$$

The first term is $O_p(M^{-1/2})$ by the previous step, while the second term is

$$(\widehat{\lambda}_M - \lambda_M^\dagger)\widehat{\mu}_M(c) = O_p(M^{-1/2})\, O_p(M^{-1/2}) = O_p(M^{-1}).$$

Therefore

$$\widehat{R}_M^{\mathrm{PPAT}}(\widehat{\lambda}_M) - R_M = O_p(M^{-1/2}),$$

which proves part 1.

It remains to prove asymptotic normality. Since $|\lambda_M^\dagger| = O(1)$, the additional $(2+\delta)$-moment assumptions control the corresponding moment of $z^\dagger$: by $|a-b|^{2+\delta} \le 2^{1+\delta}(|a|^{2+\delta} + |b|^{2+\delta})$,

$$\frac{1}{N_M}\sum_{i=1}^{N_M} |z_{M,i}^\dagger|^{2+\delta} \le C\left[\frac{1}{N_M}\sum_{i=1}^{N_M} |\ell_i|^{2+\delta} + |\lambda_M^\dagger|^{2+\delta}\frac{1}{N_M}\sum_{i=1}^{N_M} |c_{M,i}|^{2+\delta}\right] = O(1)$$

for a finite constant $C$. The fixed array $z^\dagger$ thus satisfies the hypotheses of the LURE central limit theorem (Thm. E.13), and, using the assumed variance stabilisation $\frac{1}{M}\sum_{m=1}^{M} \gamma_{M,m}^2 s_{M,m}^2(z^\dagger) \to_p \sigma_\dagger^2$, we obtain

$$\sqrt{M}\{\widehat{\mu}_M(z^\dagger) - R_M\} \Rightarrow N(0, \sigma_\dagger^2).$$

Finally, we show that replacing $\lambda_M^\dagger$ by $\widehat{\lambda}_M$ is first-order negligible. From the decomposition above,

$$\sqrt{M}\left\{\widehat{R}_M^{\mathrm{PPAT}}(\widehat{\lambda}_M) - \widehat{\mu}_M(z^\dagger)\right\} = -(\widehat{\lambda}_M - \lambda_M^\dagger)\sqrt{M}\,\widehat{\mu}_M(c).$$

We have shown $\widehat{\lambda}_M - \lambda_M^\dagger = O_p(M^{-1/2}) = o_p(1)$, while $\sqrt{M}\,\widehat{\mu}_M(c) = O_p(1)$; hence the right-hand side is $o_p(1)$, that is,

$$\sqrt{M}\left\{\widehat{R}_M^{\mathrm{PPAT}}(\widehat{\lambda}_M) - \widehat{\mu}_M(z^\dagger)\right\} = o_p(1).$$

Combining this asymptotic equivalence with the central limit theorem for $\widehat{\mu}_M(z^\dagger)$ and applying Slutsky's theorem yields

$$\sqrt{M}\left\{\widehat{R}_M^{\mathrm{PPAT}}(\widehat{\lambda}_M) - R_M\right\} \Rightarrow N(0, \sigma_\dagger^2),$$

which proves part 2. □

The preceding theorem gives a limiting normal distribution with generally unknown variance. We now show that this variance can be replaced by an estimate computed from the acquired labels, while still yielding asymptotically valid confidence intervals. This allows us to obtain approximate confidence intervals for our plug-in PPAT estimator.

**Corollary E.19** (Studentised plug-in PPAT CLT)**.** *Assume the conditions of Thm. E.18, and in addition that*

$$\frac{1}{N_M}\sum_{i=1}^{N_M} |\ell_i|^4 = O(1), \qquad \frac{1}{N_M}\sum_{i=1}^{N_M} |c_{M,i}|^4 = O(1).$$

*For $\lambda \in \mathbb{R}$, define the residualised array $z_{M,i}(\lambda) = \ell_i - \lambda c_{M,i}$ with running average*

$$A_{M,m}(\lambda) = \frac{1}{N_M}\left\{\frac{z_{M,I_{M,m}}(\lambda)}{Q_{M,m}(I_{M,m})} + \sum_{t=1}^{m-1} z_{M,I_{M,t}}(\lambda)\right\},$$

*and let*

$$\widehat{\sigma}_M^2(\widehat{\lambda}_M) = \frac{1}{M}\sum_{m=1}^{M} \gamma_{M,m}^2\left\{A_{M,m}(\widehat{\lambda}_M) - \widehat{R}_M^{\mathrm{PPAT}}(\widehat{\lambda}_M)\right\}^2.$$

*Then*

$$\widehat{\sigma}_M^2(\widehat{\lambda}_M) \to_p \sigma_\dagger^2,$$

*and hence*

$$\frac{\sqrt{M}\left\{\widehat{R}_M^{\mathrm{PPAT}}(\widehat{\lambda}_M) - R_M\right\}}{\widehat{\sigma}_M(\widehat{\lambda}_M)} \Rightarrow N(0,1).$$

*Proof.* Recall from Thm. E.18 the oracle residualised array $z^\dagger_{M,i} = \ell_i - \lambda^\dagger_M c_{M,i}$, together with the facts $|\lambda^\dagger_M| = O(1)$ and $\widehat{\lambda}_M - \lambda^\dagger_M = o_p(1)$. Evaluated at the deterministic oracle coefficient, $\widehat{\sigma}^2_M(\lambda^\dagger_M)$ is precisely the LURE variance estimator for the fixed array $z^\dagger$. The variance-estimation result of Thm. E.15 therefore applies: its moment condition holds because $|\lambda^\dagger_M| = O(1)$ and the fourth-moment assumptions give

$$\frac{1}{N_M}\sum_{i=1}^{N_M} |z^\dagger_{M,i}|^4 = O(1),$$

This gives $\widehat{\sigma}^2_M(\lambda^\dagger_M) \to_p \sigma^2_\dagger$.

It remains to replace $\lambda^\dagger_M$ with $\widehat{\lambda}_M$. Both $A_{M,m}(\lambda)$ and $\widehat{R}^{\mathrm{PPAT}}_M(\lambda) = \widehat{\mu}_M(\ell) - \lambda\widehat{\mu}_M(c)$ are affine in $\lambda$, so

$$A_{M,m}(\widehat{\lambda}_M) - \widehat{R}^{\mathrm{PPAT}}_M(\widehat{\lambda}_M) = \left\{A_{M,m}(\lambda^\dagger_M) - \widehat{R}^{\mathrm{PPAT}}_M(\lambda^\dagger_M)\right\} - (\widehat{\lambda}_M - \lambda^\dagger_M)\left\{A_{M,m}(c) - \widehat{\mu}_M(c)\right\},$$

where $A_{M,m}(c)$ is the running average of the array $c_{M,\cdot}$. Substituting this identity into $\widehat{\sigma}^2_M(\widehat{\lambda}_M)$ and expanding, every term other than $\widehat{\sigma}^2_M(\lambda^\dagger_M)$ carries a factor of $\widehat{\lambda}_M - \lambda^\dagger_M$. Since $\widehat{\lambda}_M - \lambda^\dagger_M = o_p(1)$, while $\widehat{\sigma}^2_M(\lambda^\dagger_M)$ and the empirical quadratic average

$$\frac{1}{M}\sum_{m=1}^{M} \gamma^2_{M,m}\left\{A_{M,m}(c) - \widehat{\mu}_M(c)\right\}^2$$

are both $O_p(1)$ (the latter by the fourth-moment condition on $c$), these terms are $o_p(1)$. Hence

$$\widehat{\sigma}^2_M(\widehat{\lambda}_M) - \widehat{\sigma}^2_M(\lambda^\dagger_M) = o_p(1), \qquad \text{and therefore} \qquad \widehat{\sigma}^2_M(\widehat{\lambda}_M) \to_p \sigma^2_\dagger.$$

The studentised CLT then follows from Thm. E.18 and Slutsky's theorem. □

We conclude this section by quantifying the finite–sample bias introduced by using $\widehat{\lambda}_M$. For the following two propositions, assume $1 \le M < N_M$ and $D_M > 0$.

**Proposition E.20** (Exact plug-in bias)**.** *For any APP in the sense of Def. E.1, the bias of the plug–in estimator is given by*

$$\begin{aligned}
&\mathbb{E}[\widehat{R}^{\mathrm{PPAT}}_M(\widehat{\lambda}_M)] - R_M \\
&\quad = -\frac{\mathrm{Cov}\{\widehat{G}_M, \widehat{\mu}_M(c)\}}{D_M} \\
&\quad = -\frac{1}{D_M M^2 N_M^2}\sum_{m=1}^{M} \gamma^2_{M,m}\,\mathbb{E}\left[\sum_{i\in S_{M,m}} \frac{\ell_i c^2_{M,i}}{Q_{M,m}(i)} - \left(\sum_{i\in S_{M,m}} \ell_i c_{M,i}\right)\left(\sum_{i\in S_{M,m}} c_{M,i}\right)\right].
\end{aligned} \tag{30}$$

*Proof.* By Prop. E.6, $\mathbb{E}[\widehat{\mu}_M(\ell)] = R_M$ and $\mathbb{E}[\widehat{\mu}_M(c)] = 0$. Since $\widehat{\lambda}_M = \widehat{G}_M/D_M$, we have

$$\mathbb{E}[\widehat{R}^{\mathrm{PPAT}}_M(\widehat{\lambda}_M)] - R_M = -\frac{\mathbb{E}[\widehat{G}_M\widehat{\mu}_M(c)]}{D_M} = -\frac{\mathrm{Cov}\{\widehat{G}_M, \widehat{\mu}_M(c)\}}{D_M}.$$

This gives us the first equality. To prove the second equality, first note that, by linearity of $\widehat{\mu}_M$, we have

$$\mathrm{Cov}\{\widehat{G}_M, \widehat{\mu}_M(c)\} = \frac{1}{4}\left[\mathrm{Var}\{\widehat{\mu}_M(\ell c + c)\} - \mathrm{Var}\{\widehat{\mu}_M(\ell c - c)\}\right].$$

For either sign, evaluating the conditional variance in Prop. E.6 gives

$$\mathrm{Var}\{\widehat{\mu}_M(\ell c \pm c)\} = \frac{1}{M^2 N_M^2}\sum_{m=1}^{M} \gamma^2_{M,m}\,\mathbb{E}\left[\sum_{i\in S_{M,m}} \frac{(\ell_i c_{M,i} \pm c_{M,i})^2}{Q_{M,m}(i)} - \left(\sum_{i\in S_{M,m}} (\ell_i c_{M,i} \pm c_{M,i})\right)^2\right].$$

Subtracting these two expressions and using $(u+v)^2-(u-v)^2=4uv$ therefore gives

$$\mathrm{Cov}\{\widehat{G}_M,\widehat{\mu}_M(c)\}=\frac{1}{M^2N_M^2}\sum_{m=1}^{M}\gamma_{M,m}^2\,\mathbb{E}\left[\sum_{i\in S_{M,m}}\frac{\ell_i c_{M,i}^2}{Q_{M,m}(i)}-\left(\sum_{i\in S_{M,m}}\ell_i c_{M,i}\right)\left(\sum_{i\in S_{M,m}}c_{M,i}\right)\right].$$

Substituting this into the preceding bias identity proves (30). □

**Proposition E.21** (Bias bound and rate)**.** *Suppose that, for some $\beta>0$, $Q_{M,m}(i)\ge\beta/N_M$ for all $m\le M$ and $i\in S_{M,m}$. Then*

$$\begin{aligned}&\left|\mathbb{E}[\widehat{R}_M^{\mathrm{PPAT}}(\widehat{\lambda}_M)]-R_M\right|\\&\le\frac{\sum_{m=1}^{M}\gamma_{M,m}^2}{\beta M^2}\sqrt{\frac{N_M^{-1}\sum_{i=1}^{N_M}\ell_i^2c_{M,i}^2}{D_M}}\\&\le\frac{N_M}{\beta M(N_M-M+1)}\sqrt{\frac{N_M^{-1}\sum_{i=1}^{N_M}\ell_i^2c_{M,i}^2}{D_M}}.\end{aligned}\tag{31}$$

*In particular, if $N_M/M\to\alpha>1$, the overlap constant $\beta$ is independent of $M$, and, for some $d>0$ and all sufficiently large $M$,*

$$D_M\ge d,\qquad\frac{1}{N_M}\sum_{i=1}^{N_M}\ell_i^2c_{M,i}^2=O(1),$$

*then*

$$\mathbb{E}[\widehat{R}_M^{\mathrm{PPAT}}(\widehat{\lambda}_M)]-R_M=O(M^{-1}),$$

$$\sqrt{M}\left\{\mathbb{E}[\widehat{R}_M^{\mathrm{PPAT}}(\widehat{\lambda}_M)]-R_M\right\}\longrightarrow 0.$$

*Proof.* As in the proof of Prop. E.9, Prop. E.6 and overlap give, for any fixed pool array $a_{M,1:N_M}$,

$$\mathrm{Var}\{\widehat{\mu}_M(a)\}\le\frac{\sum_{m=1}^{M}\gamma_{M,m}^2}{\beta M^2}\frac{1}{N_M}\sum_{i=1}^{N_M}a_{M,i}^2.$$

Note that this bound applies to $\widehat{G}_M=\widehat{\mu}_M(\ell c)$ and $\widehat{\mu}_M(c)$. Now, using the exact bias identity in Prop. E.20 and Cauchy–Schwarz, we have

$$\begin{aligned}&\left|\mathbb{E}[\widehat{R}_M^{\mathrm{PPAT}}(\widehat{\lambda}_M)]-R_M\right|\\&\le\frac{\sqrt{\mathrm{Var}(\widehat{G}_M)\,\mathrm{Var}\{\widehat{\mu}_M(c)\}}}{D_M}\\&\le\frac{\sum_{m=1}^{M}\gamma_{M,m}^2}{\beta M^2}\sqrt{\frac{N_M^{-1}\sum_{i=1}^{N_M}\ell_i^2c_{M,i}^2}{D_M}}.\end{aligned}$$

Using the weight bound established in the proof of Prop. E.9 gives

$$\sum_{m=1}^{M}\gamma_{M,m}^2\le\frac{N_MM}{N_M-M+1},$$

which proves the second inequality in (31). The rate follows as $N_M/(N_M-M+1)$ is bounded under the assumption $N_M/M\to\alpha>1$. □

Note that the assumptions of Thm. E.18 (1) already imply this rate. Moreover, the assumptions of Thm. 4.1, together with $D_M\ge d>0$, also suffice, as

$$\frac{1}{N_M}\sum_{i=1}^{N_M}c_{M,i}^4\le\frac{16}{N_M}\sum_{i=1}^{N_M}\widetilde{\ell}_i^4,$$

$$\frac{1}{N_M}\sum_{i=1}^{N_M}\ell_i^2c_{M,i}^2\le\sqrt{\left(\frac{1}{N_M}\sum_{i=1}^{N_M}\ell_i^4\right)\left(\frac{1}{N_M}\sum_{i=1}^{N_M}c_{M,i}^4\right)}.$$

### E.5.5 Discussion of Assumptions

Thms. E.12, E.13 and E.15 rely on a number of assumptions, which we now examine one at a time. Broadly, they fall into four groups: the finite–population asymptotic regime, two conditions on the active proposal (admissibility and uniform overlap), a family of moment conditions on the loss array, and the stabilisation of the predictable quadratic variation. As we argue below, the first three are either standard, directly enforceable, or mild conditions on the fixed test pool that involve no distributional assumptions; only the last is genuinely substantive. Throughout, recall that these results are stated for the general triangular array $\xi_{M,i}$ and specialise to LURE ($\xi_{M,i} = \ell_i$) and PPAT ($\xi_{M,i} = \zeta_{M,i}(\lambda) = \ell_i - \lambda(\tilde{\ell}_i - \tilde{R}_M)$).

**Finite–population regime.** All three results are stated in the finite–population asymptotic regime, in which the pool size $N_M$ and the budget $M$ grow together with $N_M/M \to \alpha$ for some $\alpha > 1$. This is the standard regime for without–replacement sampling from a finite population [22], and it simply says that we label a fixed fraction $M/N_M \to 1/\alpha \in (0,1)$ of the pool. The requirement $\alpha > 1$ rules out the degenerate case in which we label (essentially) the entire pool.

**Uniform overlap.** This is a positivity (overlap) condition requiring that, at every round, each remaining point retains at least a fixed fraction of uniform sampling mass. It is the finite–population analogue of the overlap/positivity assumption that is ubiquitous in importance sampling, inverse–probability weighting, and off–policy evaluation, and it plays exactly the same role here: it keeps the importance weights bounded. Crucially, this condition is not merely plausible but is enforced by construction in our experiments, as described in §3.2.

**Moment conditions on the loss array.** The three results require control of, respectively, the second, $(2+\delta)$–th, and fourth finite–pool moments of the array:

$$\underbrace{\tfrac{1}{N_M}\sum_{i=1}^{N_M}\xi_{M,i}^2 = O(1),}_{\text{consistency (Thm. E.12)}} \qquad \underbrace{\tfrac{1}{N_M}\sum_{i=1}^{N_M}|\xi_{M,i}|^{2+\delta} = O(1),}_{\text{normality (Thm. E.13)}} \qquad \underbrace{\tfrac{1}{N_M}\sum_{i=1}^{N_M}|\xi_{M,i}|^{4} = O(1)\,.}_{\text{variance/CIs (Thm. E.15)}}$$

The single most important point about these conditions is that they are statements about the empirical moments of a fixed, deterministic collection of losses: because we work conditionally on the test pool, they involve no distributional or i.i.d. assumption on the data whatsoever. They simply require that the average magnitude of the (residualised) losses does not blow up as the pool grows.

For PPAT, the array $\xi_{M,i} = \zeta_{M,i}(\lambda)$ is a linear combination of the true loss $\ell_i$ and the centred proxy loss $c_{M,i} = \widetilde{\ell}_i - \widetilde{R}_M$. Since centring does not increase moments beyond a constant factor, each of the above conditions reduces to the same bounded–moment requirement on the true losses $\ell_i$ and the proxy losses $\tilde{\ell}_i$ separately – for instance, $N_M^{-1}\sum_i \ell_i^4 = O(1)$ and $N_M^{-1}\sum_i \tilde{\ell}_i^4 = O(1)$ for the fourth–moment condition. These are easy to satisfy. For any *bounded* loss – for example the 0–1 loss, or any loss on a compact label space – all finite–pool moments are automatically bounded and the conditions hold trivially. For *unbounded* losses such as squared error or cross–entropy, they hold provided the losses do not develop pool–growing heavy tails, i.e. provided no vanishingly small subset of points carries losses large enough to dominate the average $p$–th power; in practice, this is a weak requirement.

**Stabilisation of the predictable quadratic variation.** The asymptotic normality and variance–estimation results (Thms. E.13 and E.15) additionally require that the predictable quadratic variation $\sigma_M^2 = \frac{1}{M}\sum_{m=1}^{M}\gamma_{M,m}^2\, s_{M,m}^2$ converge in probability to a finite, strictly positive constant $\sigma^2$. This is the one genuinely non–trivial assumption that we require, which we now discuss below.

Firstly, we note that it is the standard stabilisation condition for a martingale CLT [27, Chapter 3]. The centred, rescaled error $\sqrt{M}(\widehat{R}_M - R_M)$ is a sum of martingale differences, and the fluctuations of such a sum are governed by its accumulated conditional variance $\sigma_M^2$; the CLT holds precisely when this accumulated variance stabilises.

Secondly, the finiteness of $\sigma^2$ is essentially free: combining the overlap bound with the second–moment condition gives $s_{M,m}^2 \le \beta^{-1} N_M^{-1}\sum_i \xi_{M,i}^2$, and since $\gamma_{M,m}$ is uniformly bounded , $\sigma_M^2$ is

uniformly bounded above. Moreover, the positivity of $\sigma^2 > 0$ asks that the estimator have genuinely non–degenerate fluctuations at the $\sqrt{M}$ scale. It fails only in degenerate situations, for example if the residualised losses became asymptotically constant across the pool. The convergence requirement, however, is more substantive and generally does not follow from pool–level moment bounds alone, except in simpler cases such as non-adaptive proposals or random sampling without replacement. While it is difficult to establish this convergence for arbitrary fully adaptive proposals, our coverage experiments in §5.4 provide indirect empirical support for it: the PPAT intervals attain the nominal coverage level, and do so at least as quickly as, and often faster than, the competing methods. Such behaviour would be difficult to achieve if $\sigma_M^2$ failed to stabilise. We also note that while the random and LURE baselines sometimes fall short of nominal coverage at the budgets considered, we expect this to improve with an increased labelling budget.

## NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Yes, the paper extensively supports its claims with experiments on both regression and classification datasets.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: There is a separate section in the Appendix to discuss the limitations of the work.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: Yes. All proofs are provided in the Appendix and all the assumptions are stated explicitly in the Appendix.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Yes. All experimental details are included in the main body and Appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Code is available at: https://github.com/kiaashour/PPAT.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Yes. These are all detailed in section of the Appendix discussing the setup of our experiments.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We follow previous work and report the median squared error. Moreover, as our method provides (approximate) confidence intervals, we report the width of these confidence intervals in the Appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Yes, this is given in the Appendix where we discuss the details of our experimental setup.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Yes, the paper conforms to the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: This paper develops a statistical method for label-efficient estimation of a model's test risk, and is foundational research that is not tied to any particular application or deployment. Since the method only changes how existing models are evaluated, and we release no new models or datasets, we do not foresee any direct path to negative societal impacts.

    Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper poses no such risk.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: Yes, correct accreditation is provided in the text describing the experimental setup. Moreover, there is a dedicated licenses section.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.

- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: No new assets are released.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: [N/A]

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: [N/A]

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.