

LENS: LEVEL-SET ENSEMBLE FOR *post-hoc* UNCERTAINTY QUANTIFICATION

Anonymous authors
Paper under double-blind review

ABSTRACT

To exploit the nonlinear geometry of the empirical loss landscape in parameter space, we introduce *Level-set ENSemble (LENS)*, an ensemble-based *post-hoc* uncertainty quantification method. Rather than imposing a parametric approximation to the posterior or restricting uncertainty exploration to a linear subspace, *LENS* constructs an ensemble of parameter configurations constrained to lie near a fixed loss level set. This set contains models with empirical loss comparable to the trained model but potentially different predictions. The resulting model ensemble yields uncertainty estimates derived from structured, non-linear variation in parameter space. The proposed method is architecture-agnostic and does not require retraining. Across diverse prediction tasks spanning text, image, and tabular data, *LENS* improves uncertainty estimation and calibration over strong *post-hoc* methods, while maintaining predictive accuracy. Our results establish loss level sets as a principled and practical route to exploring non-linear uncertainty structure in deep learning models.

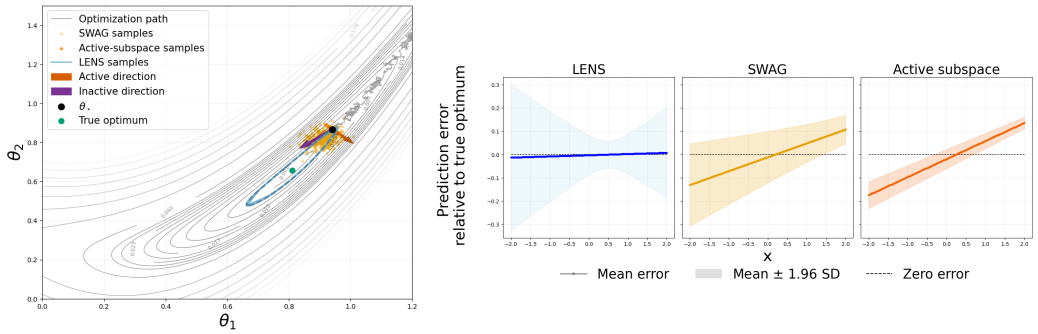
1 INTRODUCTION

Reliable decision making is a central requirement for the deployment of machine learning (ML) models in real-world settings that require models providing accurate predictions while communicating their uncertainty (Begoli et al., 2019; Blasco et al., 2024; Gawlikowski et al., 2023; Kompa et al., 2021). Uncertainty quantification (UQ) addresses this need by providing such measures of confidence supporting informed decisions particularly under ambiguity and distribution shift (Kendall & Gal, 2017; Lakshminarayanan et al., 2017; Hüllermeier & Waegeman, 2021; Abdar et al., 2021).

Bayesian methods provide a natural framework for uncertainty quantification by treating model parameters as random variables and updating prior beliefs with observed data to obtain a posterior distribution (Blundell et al., 2015; Zhou et al., 2020; Daxberger et al., 2021; Zhang et al., 2020; Blei et al., 2017; Maddox et al., 2019). These methods have led to significant progress in calibrated prediction and epistemic uncertainty estimation. However, exact Bayesian inference is generally intractable for modern neural networks Izmailov et al. (2021), and practical methods must introduce approximations. These approximations often impose a distributional family, a local geometry, or a sampling heuristic in parameter space. As a result, the resulting uncertainty estimates may depend not only on the data and model, but also on the particular approximation used. For example, SWAG Maddox et al. (2019) fits a low-rank plus diagonal Gaussian approximation to the trajectory of stochastic gradient descent iterates near a trained solution. This makes uncertainty estimation practical and inexpensive, but the resulting samples remain tied to a localized Gaussian description of the training trajectory.

A complementary line of work studies low-dimensional sensitivity structure through active subspaces and related gradient-based dimension reduction methods (Russi, 2010; Constantine et al., 2014; 2016; Cui et al., 2020; Zahm et al., 2022; Jantre et al., 2024; Abeer et al., 2024; 2026). In a posterior or UQ setting, this can identify a low-dimensional sensitivity subspace in which uncertainty can be modeled more cheaply. This perspective is valuable, especially in high-dimensional parameterizations. However, the representation is still based on a fixed linear subspace, typically estimated

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



(a) Parameter space (b) Prediction-space error

Figure 1: A two-sample nonlinear least-squares illustration: a) the contours show $L(\theta)$ and the target level set through $\theta_\star$. b) prediction error relative to the true-optimum prediction.

from gradient-based sensitivity information. Therefore, it does not follow nonlinear structures that commonly arise in neural network loss landscapes.

Figure 1 illustrates this distinction in a two-dimensional toy least-squares problem. The example is not intended as a learning benchmark; rather, it isolates the *post-hoc* sampling geometry after a reference parameter has been obtained. This mirrors the neural-network setting, where stochastic optimization typically returns a trained parameter $\theta_\star$ in a low-loss but not necessarily globally optimal region of the landscape. Although intentionally simple, this loss has a curved low-loss region. Starting from the same stochastic-optimization reference point $\theta_\star$, SWAG samples form a localized cloud around the optimization trajectory, while active-subspace samples are organized by a fixed linear sensitivity basis. Details are provided in the Appendix H.

In this work, we propose *Level-set ENSemble (LENS)*, a *post-hoc* uncertainty quantification method guided directly by the geometry of the loss landscape. Rather than fitting a localized Gaussian approximation or restricting exploration to a fixed linear sensitivity subspace, our method collects parameter configurations that remain near a prescribed loss level, as shown in Figure 1 where level-set samples explicitly track the curved contour $L(\theta) \approx L(\theta_\star)$. These samples correspond to alternative models with comparable empirical performance, but potentially different predictions on individual inputs. This disagreement is informative: inputs for which many level-set models agree can be treated as more reliable, while inputs that induce disagreement reveal regions of higher uncertainty.

Following the ensemble-based view that disagreement among plausible predictors can signal uncertainty Lakshminarayanan et al. (2017), by moving along nonlinear level-set contours, *LENS* constructs a collection of models that respects the trained objective while exposing predictive variability. This provides a geometry-aware alternative to perturbation-based, Gaussian, or fixed-subspace uncertainty estimates.

Our contributions are summarized as follows:

- We introduce *LENS*, a geometry-guided *post-hoc* uncertainty quantification method that explores nonlinear level-set structure in the loss landscape, moving beyond local Gaussian and fixed linear subspace approximations.
- We develop a practical procedure that constructs ensembles of comparable-loss models with potentially different predictions, allowing the use of structured predictive disagreement as an uncertainty signal.
- We empirically demonstrate that *LENS* provides reliable uncertainty estimates, improving calibration across diverse predictive settings, including regression, image classification, and LoRA-adapted language models for reasoning benchmarks, while maintaining competitive predictive performance.

2 Related Work

Uncertainty quantification in neural networks has been studied through Bayesian, approximate Bayesian, stochastic, and ensemble-based approaches. Practical methods rely on approximations such as variational inference, Laplace approximations, stochastic-gradient MCMC, Monte Carlo

dropout, and stochastic weight averaging Gaussian methods (Goan & Fookes, 2020; Gal & Ghahramani, 2016; Blei et al., 2017; Daxberger et al., 2021; Maddox et al., 2019; Welling & Teh, 2011). Deep ensembles provide another strong baseline by training multiple independently initialized models and using their predictive disagreement as an uncertainty signal (Lakshminarayanan et al., 2017). Related trajectory-based ensemble methods define ensembles through parameter trajectories and encourage these trajectories to remain in low-loss regions (Garipov et al., 2018). These methods have been effective for calibration, uncertainty estimation, and robustness, and they motivate the broader idea that useful uncertainty signals can be obtained from multiple plausible predictors rather than a single deterministic model.

Recent work has adapted uncertainty quantification to parameter-efficient fine-tuning. Low-rank adaptation (LoRA) provides an efficient way to fine-tune large language models by restricting updates to low-rank adapter parameters (Hu et al., 2022). Bayesian and uncertainty-aware LoRA methods apply approximate Bayesian inference or stochastic prediction mechanisms in the adapter space (Yang et al., 2024; Wang et al., 2024; Rahmati et al., 2025). Other curvature-aware LoRA methods use second-order or Fisher/Hessian-inspired information to allocate capacity, guide rank scheduling, or constrain updates through trust-region mechanisms (Wang et al., 2026). Recent Bayesian LoRA focuses on low-loss adapter-space defined by fine-tuned anchors or learned control points Dold et al. (2026). Together, these works show that adapter-space uncertainty and geometry are important for calibrated and robust fine-tuning of large models.

In neural network analysis, prior works have studied how level sets of network outputs vary with respect to inputs and how neural tuning landscapes form structured level sets over image space (Atzmon et al., 2019; Wang & Ponce, 2023). In interpretability, the loss kernel uses low-loss-preserving parameter perturbations around a trained network to compute covariances between per-example losses, yielding a kernel for data attribution (Adam et al., 2025). Level-set hit-and-run has been developed for efficient sampling within superlevel sets following high-dimensional distributions (Jensen & Foster, 2014). In optimization, symmetry teleportation uses known loss-invariant group actions to move parameters along exact loss-preserving symmetry orbits before continuing gradient-based training (Zhao et al., 2022). These works demonstrate the utility of level-set structures, but primarily for input-space analysis, loss-covariance estimation, density sampling, or symmetry-aware optimization.

Building on these efforts, *LENS* constructs a predictive ensemble for UQ through approximately loss-preserving movements in parameter space.

3 LENS

LENS is a *post-hoc* uncertainty quantification (UQ) method that explores the geometry of the loss landscape. Starting from a trained model, *LENS* proposes tangent directions approximately orthogonal to the loss gradient and then corrects proposals back toward the target loss level. The resulting parameter vectors have comparable empirical fit but may induce different predictions.

3.1 Problem Setting

Let $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^n$ denote the data used to define the loss landscape, and let f_θ be a model parametrized by $\theta \in \mathbb{R}^p$, trained with the corresponding loss function measuring empirical fit:

$$L(\theta) = \frac{1}{n} \sum_{i=1}^{n} \ell(f_\theta(x_i), y_i),$$

where ℓ is the loss of the corresponding prediction task. Suppose training produces a parameter vector $\theta_\star$. We define the target loss level as $\ell_\star = L(\theta_\star)$. The exact level set through $\theta_\star$ is

$$\mathcal{M}_{\ell_\star} = \{\theta \in \mathbb{R}^p : L(\theta) = \ell_\star\}.$$

Assume $\ell_\star > 0$. We measure deviation from the target loss level using the relative residual

$$r_{\theta_\star}(\theta) = \frac{L(\theta) - \ell_\star}{\ell_\star}.$$

Definition 3.1 (Relative level-set tube) *For a tolerance $\varepsilon > 0$, the relative level-set tube around the trained solution is*

$$\mathcal{T}_\varepsilon(\ell_\star) = \{\theta \in \mathbb{R}^p : |r_{\theta_\star}(\theta)| \le \varepsilon\}.$$

We call parameters in $\mathcal{T}_\varepsilon(\ell_\star)$ ε-loss-plausible relative to $\theta_\star$.

Definition 3.2 (Level-set ensemble) *A* LENS *ensemble of size N and tolerance ε is a finite set*

$$\mathcal{S}_{\varepsilon,N} = \{\theta^{(1)}, \ldots, \theta^{(N)}\} \subset \mathcal{T}_\varepsilon(\ell_\star),$$

whose members define alternative predictors with comparable empirical loss.

Predictive variation across this ensemble measures sensitivity to approximately loss-preserving parameter changes. *LENS* is therefore not formulated as Markov chain Monte Carlo or as an approximation to a prescribed Bayesian posterior. It instead uses constrained model variation to construct an ensemble for prediction and uncertainty estimation.

3.2 Loss Level Sets as Uncertainty Sets

Modern AI/ML models often admit many parameter configurations with similar training/validation loss values. *LENS* aims to construct a level-set ensemble, as in Definition 3.2, whose members lie in the relative tube of Definition 3.1 while remaining diverse enough to reveal predictive instability.

Proposition 3.3 (Comparable empirical fit) *If $\theta \in \mathcal{T}_\varepsilon(\ell_\star)$, then*

$$(1-\varepsilon)L(\theta_\star) \le L(\theta) \le (1+\varepsilon)L(\theta_\star).$$

Proposition 3.3 formalizes that *LENS* samples remain plausible under the training objective. The uncertainty signal therefore comes from disagreement among predictors with comparable empirical fit, rather than arbitrary degradation from the trained model. This differs from posterior approximations such as SWAG methods, which impose a local parametric distribution around $\theta_\star$, or fixed-subspace methods which restrict variation to a predetermined linear subspace.

3.3 Tangent Geometry of the Level Set

For a current point θ with $\nabla L(\theta) \neq 0$, consider the local level set through θ,

$$\mathcal{M}_{L(\theta)} = \{\theta' \in \mathbb{R}^p : L(\theta') = L(\theta)\}.$$

Its tangent space is

$$T_\theta \mathcal{M}_{L(\theta)} = \{v \in \mathbb{R}^p : \nabla L(\theta)^\top v = 0\}.$$

Given an arbitrary direction $d \in \mathbb{R}^p$ and gradient $g = \nabla L(\theta)$, we remove the component of d in the gradient direction:

$$\widetilde{v} = d - \frac{d^\top g}{\|g\|_2^2} g.$$

The normalized tangent direction is then $v = \widetilde{v}/\|\widetilde{v}\|_2$. By construction, $g^\top v = 0$.

Proposition 3.4 (First-order loss preservation) *Assume L is twice continuously differentiable near θ, with $\nabla L(\theta) \neq 0$[1]. If $v \in T_\theta \mathcal{M}_{L(\theta)}$ and $\|v\|_2 = 1$, then*

$$L(\theta + \alpha v) = L(\theta) + \frac{\alpha^2}{2} v^\top \nabla^2 L(\theta) v + o(\alpha^2).$$

In particular, parameter perturbations along the tangent direction preserve the loss to first order.

Proposition 3.4 motivates the following predictor parameter sampling step in *LENS*. Because finite steps can still drift due to curvature, *LENS* follows tangent proposals with a normal correction back toward the target tube.

[1] In our experiments, L is evaluated on a validation split disjoint from the training data used to obtain $\theta_\star$. Consequently, approximate stationarity of the training objective does not imply that $\nabla L(\theta_\star) = 0$ for the construction loss.

3.4 PREDICTOR-CORRECTOR

Each *LENS* update has two parts. First, a *predictor* step moves along a tangent direction:

$$\theta_{\text{pred}} = \theta + \alpha v, \qquad \alpha \sim \text{Unif}(\alpha_{\min}, \alpha_{\max}).$$

Second, a *corrector* step moves back toward the target loss level. A first-order approximation gives

$$L(\theta + \Delta\theta) \approx L(\theta) + \nabla L(\theta)^\top \Delta\theta.$$

The gradient gives the local orthogonal direction to the level set. We use a damped orthogonal correction proportional to the relative residual:

$$\Delta\theta = -\eta \frac{r_{\theta_\star}(\theta)}{\|\nabla L(\theta)\|_2^2} \nabla L(\theta).$$

where $\eta > 0$ is a correction step. This update helps move the proposals toward the target loss level along the local normal direction of the level set. For numerical stability, we cap the correction norm:

$$\Delta\theta \leftarrow \min\left\{1, \frac{s_{\max}}{\|\Delta\theta\|_2}\right\} \Delta\theta.$$

The corrector is a bounded feasibility step, that after at most M iterations, a proposal is accepted only if it satisfies $|r_{\theta_\star}(\theta)| \leq \varepsilon$, and is otherwise rejected. Appendix C details how the required correction $\Delta\theta$ relates to the curvature induced loss drift of a finite tangent step.

3.5 PRACTICAL REFINEMENTS

The mathematical target is the relaxed level set $\mathcal{T}_\varepsilon(\ell_\star)$. In practice, we use several refinements to improve exploration and stability.

First, we initialize two chains from $\theta_\star$ in opposite tangent directions to explore both local level-set orientations. Given a random seed direction d_0, we compute a tangent direction v_0 at $\theta_\star$ and initialize

$$(\theta^{(0)}, v^{(0)}) = (\theta_\star, v_0), \qquad (\theta^{(1)}, v^{(1)}) = (\theta_\star, -v_0).$$

Second, each chain uses noisy tangent continuation. Instead of drawing an independent direction at every step, we perturb the previous direction:

$$\widehat{v} = v_{\text{prev}} + \sigma_d \xi, \qquad \xi \sim \mathcal{N}(0, I).$$

We then project $\widehat{v}$ onto the current tangent space. If the new direction reverses orientation, we flip its sign for a smoother movement along the level set while allowing stochastic exploration

$$\text{if } v^\top v_{\text{prev}} < 0, \qquad v \leftarrow -v.$$

Third, when $\|\nabla L(\theta)\|_2$ is below a numerical tolerance τ_g, the tangent projection is not reliably defined. During direction construction, we therefore use the normalized proposed direction as an exploratory fallback; such a move is not interpreted as a tangent step. During correction, a proposal whose gradient remains below the tolerance is rejected unless it already satisfies the level-set criterion. In all cases, inclusion in the final ensemble is determined by the explicit relative-loss test.

Finally, for large language models we apply *LENS* only to the trainable parameter subset, such as LoRA adapters. The pretrained backbone remains fixed. Thus, all gradients, tangent projections, and corrections are computed in the adapter parameter space.

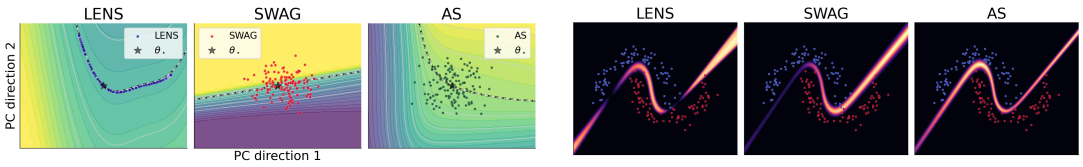
3.6 PREDICTION AND UNCERTAINTY ESTIMATION

The *LENS* prediction is obtained by model averaging over accepted samples

$$\widehat{p}(x) = \frac{1}{N} \sum_{k=1}^{N} \sigma_{\theta^{(k)}}(x) \quad \theta^{(k)} \in \mathcal{T}_\varepsilon(\ell_\star).$$

where $\sigma_{\theta^{(k)}}(x)$ denotes the prediction probability induced by the k-th model configuration.

In summary, *LENS* asks: among models that fit the objective comparably well, how stable is the prediction? If samples agree on an input, the prediction is stable under loss-preserving parameter variation. If they disagree, then the objective does not uniquely determine the model behavior for that input. *LENS* uses this structured disagreement as a *post-hoc* uncertainty signal.



(a) Parameter space (b) Prediction space

Figure 2: a) Two-moons parameter-space visualization. Samples are projected onto a two-dimensional PCA plane around the trained solution. b) Predictive disagreement on two-moons.

4 EXPERIMENTS

We evaluate *LENS* across four complementary settings with different prediction tasks and different input/output data modalities. First, a two-moons classification problem visualizes how different sampling schemes explore predictive variability in a small neural network. Second, UCI regression benchmarks test whether *LENS* ensemble-construction principle transfers beyond classification with tabular input feature vectors. Third, CIFAR-10 Krizhevsky (2009) classification with CIFAR-10-C Hendrycks & Dietterich (2019) evaluation under distribution shift in image data evaluates whether *LENS* remains effective in a realistic large-scale classification benchmark. Finally, LoRA fine-tuning of Qwen2.5-7B Qwen et al. (2025) on question-answering benchmarks tests the applicability of *LENS* to modern LLMs for *post-hoc* UQ in foundation-model adaptation. Across these settings, our main focus is not only predictive performance, but whether the resulting uncertainty estimates are calibrated and avoid overconfident predictions.

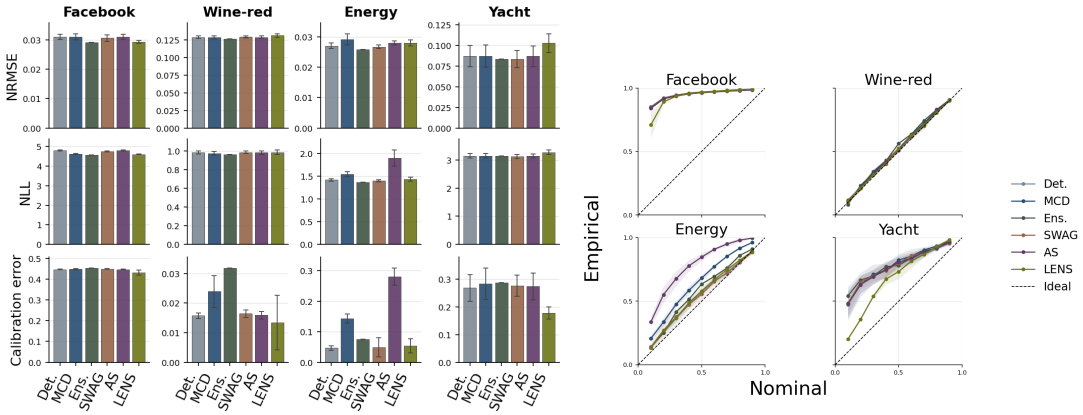
Methods. We evaluate *LENS* against a deterministic baseline and several *post-hoc* uncertainty estimation methods. In our results, **Det.** denotes the deterministic inference. **Ens.** denotes a deterministic ensemble formed from independently trained models. **MCD** applies Monte Carlo dropout at inference time. **SWAG** uses a SWAG Gaussian approximation obtained from a fixed-learning-rate refinement trajectory. **AS** denotes active-subspace sampling. Full implementation details are provided in the Appendix K.

4.1 TWO-MOONS CLASSIFICATION GEOMETRY

We first study a two-moons classification problem to visualize how parameter-space sampling affects predictive uncertainty. Although this setting is intentionally small, it extends the geometric intuition of Figure 1 to a nonlinear classification setting. Figure 2a shows the samples for each method projected onto a two-dimensional PCA plane, together with the empirical-loss landscape induced by that same plane. For each method, the PCA basis is computed from its sampled parameters and $\theta_\star$, and the loss is then evaluated on the corresponding plane. Hence, each panel should be interpreted as a method-specific cross-section of the high-dimensional landscape along the directions explored by that sampler. In this view, *LENS* visibly traverse the loss level set through the trained solution, while SWAG and AS samples explore nearby regions, reflecting the local covariance and sensitivity structures used by these methods. This visualization reinforces the interpretation of *LENS* as a geometry-guided ensemble that directly collects diverse models with comparable loss rather than imposing a parametric posterior approximation. Figure 2b shows the corresponding predictive disagreement over input space, measured by the standard deviation of $p(y = 1)$ across sampled models. For *LENS*, this disagreement arises from models constrained to have comparable predictive performance, illustrating how loss-preserving parameter variation can induce localized predictive uncertainty. In particular, *LENS* puts uncertainty where data support is weak while keeping disagreement low in well-supported regions. This pattern is less pronounced for SWAG and AS that are not constrained to explore the same loss level.

4.2 UCI REGRESSION BENCHMARKS

We further evaluate *LENS* on UCI regression benchmarks. We use four datasets: Facebook comment volume (Singh, 2015), Wine-red (Cortez et al., 2009), Energy (Tsanas & Xifara, 2012), and Yacht (Gerritsma et al., 1981). For each dataset, we train three deterministic base models and apply *post-hoc* UQ methods to the same saved runs. We report normalized root-mean-square-



(a) Performance across three deterministic base runs.

(b) Calibration curves.

Figure 3: UCI regression tasks.

	In-distribution: CIFAR-10				*Out-of-distribution: CIFAR-10-C*			
Method	Acc. (%) ↑	NLL ↓	ECE (%) ↓	Brier ↓	Acc. (%) ↑	NLL ↓	ECE (%) ↓	Brier ↓
Det.	$91.48_{\pm 0.27}$	$0.27_{\pm 0.01}$	$2.78_{\pm 0.21}$	$0.13_{\pm 0.00}$	$53.66_{\pm 0.69}$	$1.99_{\pm 0.02}$	$26.57_{\pm 0.10}$	$0.71_{\pm 0.01}$
MCD	$91.48_{\pm 0.25}$	$0.26_{\pm 0.01}$	$0.81_{\pm 0.26}$	$0.13_{\pm 0.00}$	$53.50_{\pm 0.63}$	$1.69_{\pm 0.02}$	$19.73_{\pm 0.15}$	$0.66_{\pm 0.01}$
Ens.	**92.79**	**0.22**	**0.67**	**0.11**	**55.53**	1.69	19.94	0.65
SWAG	$91.07_{\pm 0.45}$	$0.27_{\pm 0.01}$	$1.84_{\pm 0.11}$	$0.13_{\pm 0.00}$	$52.84_{\pm 1.62}$	$1.90_{\pm 0.14}$	$24.23_{\pm 2.34}$	$0.70_{\pm 0.03}$
AS	$91.50_{\pm 0.29}$	$0.26_{\pm 0.01}$	$1.92_{\pm 0.09}$	$0.13_{\pm 0.00}$	$53.74_{\pm 0.77}$	$1.88_{\pm 0.03}$	$24.11_{\pm 0.28}$	$0.69_{\pm 0.01}$
LENS	$91.00_{\pm 0.35}$	$0.27_{\pm 0.01}$	$1.22_{\pm 0.79}$	$0.13_{\pm 0.01}$	$53.60_{\pm 0.58}$	$\mathbf{1.58_{\pm 0.02}}$	$\mathbf{16.00_{\pm 1.89}}$	$\mathbf{0.64_{\pm 0.00}}$

Table 1: CIFAR-10 in-distribution and CIFAR-10-C out-of-distribution results.

error (NRMSE), Gaussian negative log-likelihood (NLL), and calibration error over central predictive intervals.

Figure 3a shows that *LENS* maintains broadly comparable predictive performance across datasets. The methods generally have similar NRMSE and NLL, expected for the regression tasks: under a Gaussian predictive model, NLL is strongly tied to the squared prediction error as well as the predicted variance. Since all *post-hoc* methods are applied to the same deterministic base models, large improvements in point-error metrics are not expected. The main distinction appears in calibration where *LENS* achieves the lowest or among the lowest calibration errors across the regression tasks, and its calibration curves are generally closer to the ideal diagonal. The calibration curves in Figure 3b provide a more detailed view. *LENS* curves are generally closer to the ideal diagonal, indicating that its derived predictive intervals better match empirical coverage.

4.3 CIFAR-10 Image Classification

To evaluate *LENS* on different prediction tasks across different data modalities, we test it on CIFAR-10 image classification with a ResNet-20 He et al. (2016) model and CIFAR-10-C for robustness evaluation under corruption shift. The results are reported over three random seeds, except for the ensemble, which averages the three independently trained models. The sampling-based methods and *LENS* use 20 predictive samples, and ECE is computed with 10 equal-width confidence bins. Additional experimental details are provided in Appendix K.

Table 1 reports in-distribution CIFAR-10 results and out-of-distribution CIFAR-10-C results. On CIFAR-10, the ensemble gives the strongest accuracy and likelihood, while MCD and *LENS* improve calibration relative to the deterministic and SWAG baselines. Under CIFAR-10-C shift, the ensemble retains the highest accuracy, but *LENS* gives the best uncertainty-quality metrics, with lower NLL, ECE, and Brier scores than all other methods. This suggests that sampling along the loss level set provides useful predictive variation under distribution shift, even when point accuracy is similar across methods.

4.4 Foundation-Model Question Answering

We finally evaluate *LENS* for LoRA-fine-tuned large language models (LLMs). We report accuracy, ECE, and NLL. As SWAG and AS are the most closely related parameter-space baselines, they form

Method	*In-Distribution*						*OOD (Smaller Shift)*		*OOD (Larger Shift)*	
	WG-S	ARC-C	ARC-E	WG-M	OBQA	BoolQ	ARC-C	ARC-E	Chem	Phy
					Acc. (%) ↑				Acc. (%) ↑	
Det.	$72.82_{\pm 2.17}$	$89.33_{\pm 0.31}$	$96.55_{\pm 0.11}$	$79.82_{\pm 0.92}$	$92.73_{\pm 1.33}$	$\mathbf{89.22_{\pm 0.07}}$	$89.14_{\pm 0.40}$	$95.55_{\pm 0.38}$	$50.67_{\pm 2.08}$	$46.41_{\pm 3.15}$
Ens.	**73.56**	89.51	**96.84**	**80.82**	**93.00**	89.11	89.08	**95.66**	**51.00**	48.04
MCD	$72.82_{\pm 2.17}$	$89.36_{\pm 0.30}$	$96.53_{\pm 0.14}$	$79.82_{\pm 0.94}$	$92.80_{\pm 1.44}$	$89.20_{\pm 0.05}$	$89.11_{\pm 0.36}$	$95.57_{\pm 0.38}$	$50.67_{\pm 1.15}$	$46.73_{\pm 2.83}$
SWAG	$72.80_{\pm 2.22}$	$89.31_{\pm 0.34}$	$96.53_{\pm 0.10}$	$79.87_{\pm 0.91}$	$92.73_{\pm 1.33}$	$89.15_{\pm 0.02}$	$89.16_{\pm 0.39}$	$95.54_{\pm 0.36}$	$\mathbf{51.00_{\pm 1.73}}$	$46.73_{\pm 2.83}$
AS_3	$72.59_{\pm 2.80}$	$89.45_{\pm 0.27}$	$96.59_{\pm 0.19}$	$79.98_{\pm 1.02}$	$92.80_{\pm 1.31}$	$89.15_{\pm 0.05}$	$89.05_{\pm 0.30}$	$95.61_{\pm 0.40}$	$50.67_{\pm 1.53}$	$48.04_{\pm 3.53}$
AS_{10}	$72.59_{\pm 2.65}$	$89.53_{\pm 0.21}$	$96.62_{\pm 0.09}$	$80.14_{\pm 0.94}$	$92.73_{\pm 1.30}$	$89.11_{\pm 0.08}$	$89.14_{\pm 0.26}$	$95.62_{\pm 0.39}$	$49.67_{\pm 1.53}$	$46.73_{\pm 6.38}$
TFB_{10}	$71.95_{\pm 2.87}$	$89.59_{\pm 0.43}$	$96.53_{\pm 0.17}$	$80.03_{\pm 0.88}$	$\mathbf{93.00_{\pm 0.72}}$	$89.17_{\pm 0.28}$	$89.14_{\pm 0.49}$	$95.54_{\pm 0.34}$	$50.00_{\pm 2.00}$	$47.39_{\pm 5.91}$
TFB_{20}	$72.06_{\pm 2.51}$	$\mathbf{89.62_{\pm 0.30}}$	$96.51_{\pm 0.21}$	$79.87_{\pm 0.65}$	$92.80_{\pm 1.00}$	$89.12_{\pm 0.11}$	$89.14_{\pm 0.51}$	$95.59_{\pm 0.38}$	$50.00_{\pm 2.00}$	$46.73_{\pm 5.03}$
$LENS_5$	$72.14_{\pm 2.95}$	$89.42_{\pm 0.23}$	$95.79_{\pm 0.55}$	$79.85_{\pm 0.56}$	$92.93_{\pm 1.10}$	$89.14_{\pm 0.12}$	$\mathbf{89.33_{\pm 0.31}}$	$95.64_{\pm 0.22}$	$\mathbf{51.00_{\pm 1.00}}$	$49.67_{\pm 3.15}$
$LENS_{10}$	$72.03_{\pm 2.92}$	$89.39_{\pm 0.34}$	$96.44_{\pm 0.16}$	$80.06_{\pm 0.84}$	$\mathbf{93.00_{\pm 0.72}}$	$89.07_{\pm 0.04}$	$89.28_{\pm 0.18}$	$95.57_{\pm 0.35}$	$50.00_{\pm 1.00}$	$\mathbf{50.33_{\pm 4.93}}$
$LENS_{20}$	$72.09_{\pm 1.89}$	$89.25_{\pm 0.37}$	$96.25_{\pm 0.15}$	$79.01_{\pm 0.39}$	$92.60_{\pm 1.40}$	$88.94_{\pm 0.34}$	$88.94_{\pm 0.47}$	$95.23_{\pm 0.27}$	$50.00_{\pm 3.61}$	$48.69_{\pm 3.15}$
					ECE (%) ↓				ECE (%) ↓	
Det.	$21.31_{\pm 3.46}$	$9.18_{\pm 0.57}$	$2.78_{\pm 0.37}$	$18.25_{\pm 0.25}$	$5.80_{\pm 1.11}$	$8.88_{\pm 0.50}$	$8.63_{\pm 0.66}$	$3.15_{\pm 0.57}$	$34.95_{\pm 0.41}$	$36.24_{\pm 4.91}$
Ens.	17.10	7.67	2.38	13.53	4.49	8.04	7.97	2.54	33.07	37.59
MCD	$21.32_{\pm 3.39}$	$9.21_{\pm 0.50}$	$2.78_{\pm 0.37}$	$18.16_{\pm 0.28}$	$5.93_{\pm 0.89}$	$8.86_{\pm 0.49}$	$8.60_{\pm 0.59}$	$3.11_{\pm 0.56}$	$34.79_{\pm 1.28}$	$36.21_{\pm 4.38}$
SWAG	$21.37_{\pm 3.41}$	$9.23_{\pm 0.54}$	$2.74_{\pm 0.39}$	$18.30_{\pm 0.33}$	$5.80_{\pm 1.10}$	$8.88_{\pm 0.43}$	$8.64_{\pm 0.72}$	$3.15_{\pm 0.54}$	$34.49_{\pm 1.44}$	$35.58_{\pm 4.89}$
AS_3	$11.28_{\pm 4.61}$	$5.16_{\pm 3.66}$	$2.69_{\pm 0.39}$	$17.60_{\pm 0.26}$	$5.83_{\pm 1.00}$	$8.64_{\pm 0.36}$	$8.59_{\pm 0.68}$	$3.03_{\pm 0.59}$	$34.29_{\pm 1.50}$	$36.26_{\pm 4.28}$
AS_{10}	$\mathbf{9.56_{\pm 2.39}}$	$9.26_{\pm 3.51}$	$2.01_{\pm 0.65}$	$16.44_{\pm 0.75}$	$4.98_{\pm 1.12}$	$8.41_{\pm 0.28}$	$4.18_{\pm 0.29}$	$5.40_{\pm 0.93}$	$25.98_{\pm 1.27}$	$29.34_{\pm 4.83}$
TFB_{10}	$15.32_{\pm 6.71}$	$7.25_{\pm 1.53}$	$\mathbf{1.91_{\pm 0.67}}$	$15.56_{\pm 0.33}$	$4.13_{\pm 0.60}$	$6.20_{\pm 0.87}$	$7.23_{\pm 1.04}$	$2.42_{\pm 0.65}$	$32.00_{\pm 1.07}$	$33.58_{\pm 4.20}$
TFB_{20}	$15.09_{\pm 6.53}$	$6.84_{\pm 1.59}$	$\mathbf{1.91_{\pm 0.73}}$	$15.49_{\pm 0.24}$	$\mathbf{4.00_{\pm 0.69}}$	$6.09_{\pm 0.78}$	$7.17_{\pm 1.11}$	$2.16_{\pm 0.89}$	$33.07_{\pm 1.96}$	$33.49_{\pm 5.56}$
$LENS_5$	$20.45_{\pm 2.70}$	$8.11_{\pm 0.29}$	$8.78_{\pm 5.78}$	$15.70_{\pm 1.04}$	$5.07_{\pm 1.00}$	$8.15_{\pm 0.49}$	$7.92_{\pm 0.10}$	$2.88_{\pm 0.20}$	$31.98_{\pm 0.14}$	$33.63_{\pm 2.95}$
$LENS_{10}$	$18.06_{\pm 2.90}$	$6.64_{\pm 0.82}$	$1.98_{\pm 2.06}$	$13.01_{\pm 1.20}$	$4.43_{\pm 0.49}$	$\mathbf{4.50_{\pm 0.67}}$	$6.46_{\pm 1.77}$	$1.91_{\pm 0.80}$	$32.07_{\pm 4.87}$	$30.16_{\pm 3.47}$
$LENS_{20}$	$9.76_{\pm 3.58}$	$\mathbf{3.97_{\pm 0.81}}$	$4.89_{\pm 2.03}$	$\mathbf{8.47_{\pm 0.08}}$	$5.81_{\pm 5.01}$	$4.63_{\pm 1.37}$	$\mathbf{3.98_{\pm 1.02}}$	$\mathbf{1.86_{\pm 1.02}}$	$\mathbf{23.89_{\pm 2.57}}$	$\mathbf{23.31_{\pm 6.52}}$
					NLL ↓				NLL ↓	
Det.	$1.04_{\pm 0.29}$	$0.61_{\pm 0.08}$	$0.20_{\pm 0.05}$	$0.99_{\pm 0.10}$	$0.32_{\pm 0.03}$	$0.46_{\pm 0.03}$	$0.48_{\pm 0.04}$	$0.19_{\pm 0.03}$	$1.90_{\pm 0.11}$	$1.72_{\pm 0.16}$
Ens.	0.82	0.52	**0.15**	0.75	**0.26**	0.42	0.42	**0.16**	1.83	1.63
MCD	$1.04_{\pm 0.29}$	$0.60_{\pm 0.08}$	$0.20_{\pm 0.05}$	$0.99_{\pm 0.10}$	$0.32_{\pm 0.03}$	$0.46_{\pm 0.03}$	$0.47_{\pm 0.04}$	$0.18_{\pm 0.03}$	$1.90_{\pm 0.11}$	$1.72_{\pm 0.16}$
SWAG	$1.04_{\pm 0.29}$	$0.61_{\pm 0.08}$	$0.20_{\pm 0.05}$	$0.99_{\pm 0.10}$	$0.32_{\pm 0.03}$	$0.46_{\pm 0.03}$	$0.48_{\pm 0.04}$	$0.19_{\pm 0.03}$	$1.90_{\pm 0.11}$	$1.72_{\pm 0.16}$
AS_3	$0.65_{\pm 0.08}$	$0.47_{\pm 0.14}$	$0.18_{\pm 0.03}$	$0.95_{\pm 0.09}$	$0.31_{\pm 0.02}$	$0.45_{\pm 0.02}$	$0.47_{\pm 0.05}$	$0.18_{\pm 0.03}$	$1.90_{\pm 0.13}$	$1.72_{\pm 0.16}$
AS_{10}	$\mathbf{0.61_{\pm 0.04}}$	$0.45_{\pm 0.04}$	$\mathbf{0.15_{\pm 0.02}}$	$0.85_{\pm 0.05}$	$0.27_{\pm 0.02}$	$0.43_{\pm 0.01}$	$0.39_{\pm 0.02}$	$0.20_{\pm 0.01}$	$\mathbf{1.40_{\pm 0.08}}$	$\mathbf{1.35_{\pm 0.02}}$
TFB_{10}	$0.83_{\pm 0.27}$	$0.49_{\pm 0.09}$	$0.16_{\pm 0.02}$	$0.81_{\pm 0.08}$	$0.28_{\pm 0.02}$	$0.37_{\pm 0.02}$	$0.42_{\pm 0.05}$	$0.17_{\pm 0.03}$	$1.79_{\pm 0.11}$	$1.59_{\pm 0.18}$
TFB_{20}	$0.82_{\pm 0.26}$	$0.47_{\pm 0.09}$	$0.16_{\pm 0.02}$	$0.78_{\pm 0.08}$	$0.27_{\pm 0.02}$	$0.35_{\pm 0.02}$	$0.42_{\pm 0.05}$	$0.17_{\pm 0.02}$	$1.76_{\pm 0.11}$	$1.58_{\pm 0.17}$
$LENS_5$	$0.98_{\pm 0.21}$	$0.52_{\pm 0.03}$	$0.26_{\pm 0.09}$	$0.87_{\pm 0.09}$	$0.29_{\pm 0.04}$	$0.42_{\pm 0.03}$	$0.44_{\pm 0.03}$	$0.17_{\pm 0.02}$	$1.88_{\pm 0.13}$	$1.63_{\pm 0.12}$
$LENS_{10}$	$0.88_{\pm 0.14}$	$0.44_{\pm 0.06}$	$0.16_{\pm 0.04}$	$0.70_{\pm 0.06}$	$\mathbf{0.26_{\pm 0.02}}$	$\mathbf{0.33_{\pm 0.01}}$	$0.41_{\pm 0.03}$	$\mathbf{0.16_{\pm 0.01}}$	$1.80_{\pm 0.06}$	$1.53_{\pm 0.05}$
$LENS_{20}$	$0.66_{\pm 0.09}$	$\mathbf{0.40_{\pm 0.03}}$	$0.18_{\pm 0.03}$	$\mathbf{0.54_{\pm 0.02}}$	$0.28_{\pm 0.07}$	$0.34_{\pm 0.03}$	$\mathbf{0.37_{\pm 0.03}}$	$0.17_{\pm 0.02}$	$1.61_{\pm 0.15}$	$1.44_{\pm 0.13}$

Table 2: In-distribution and out-of-distribution performance on reasoning tasks. OOD results are obtained from models trained on OBQA. Best and second-best values in each metric and dataset column are shown in **bold** and underlined, respectively.

our primary comparisons. We further broaden the evaluation with Training-Free Bayesianization (TFB) Shi et al. (2024), a recent *post-hoc* UQ approach designed for low-rank adapters.

Experimental setup. All experiments use Qwen2.5-7B as the base model and apply LoRA fine-tuning to the trainable adapter parameters. We report mean and standard deviation over three random seeds. We evaluate on six multiple-choice and binary question-answering benchmarks: ARC-Challenge (ARC-C), ARC-Easy (ARC-E) (Clark et al., 2018), WinoGrande-Small (WG-S), WinoGrande-Medium (WG-M) (Sakaguchi et al., 2021), OpenBookQA (OBQA) (Mihaylov et al., 2018), and BoolQ (Clark et al., 2019).

In-distribution results. Table 2 summarizes the results across datasets. AS_d ($d = 3, 10$) correspond to different active-subspace dimensions, $LENS_K$ ($K = 5, 10, 20$) indicates the use of different numbers of collected level-set configurations, and TFB_K ($K = 10, 20$) denotes predictive averaging over K Bayesian parameter samples. Deterministic LoRA and the independently trained ensemble generally preserve the strongest raw accuracy, which is expected because they either use the fine-tuned point estimate directly or average several independently trained point predictors. However, the main advantage of *LENS* appears in the probabilistic metrics. *LENS* substantially improves ECE and NLL on most datasets while keeping accuracy competitive with the original LoRA model.

The results illustrate that an increase in K often improves *LENS* calibration and NLL, although the trend is not strictly monotonic across every dataset. $LENS_{20}$ gives the strongest *LENS* results on ARC-C and WG-M, while $LENS_{10}$ performs better on other tasks for several probabilistic metrics. On WG-S, AS_{10} remains the strongest single-model method suggesting that linear sensitivity directions can capture useful uncertainty structure in certain cases. However, *LENS* gives the most consistent improvement in calibration and likelihood across the table. TFB demonstrates strong performance on some datasets such as ARC-E and OBQA, however its gains are less consistent across datasets. SWAG remains close to deterministic LoRA in most settings, indicating that the local Gaussian approximation explored by SWAG may not provide sufficiently useful predictive variation for these fine-tuned adapters. Overall these results support the central claim that *LENS* improves uncertainty quality, often offering the best or the second best probabilistic scores without requir-

ing multiple independently trained models. A decomposition of *LENS* predictive uncertainty into epistemic and aleatoric components is provided in the Appendix G.

4.5 Robustness Under Distribution Shift

We next evaluate whether the calibration and likelihood improvements persist under distribution shift. For this experiment, all models are trained on OBQA and evaluated out-of-distribution on ARC-C, ARC-E, MMLU-Chemistry, and MMLU-Physics (Hendrycks et al., 2021). ARC-C and ARC-E represent smaller shifts because they are related science question-answering datasets, while the MMLU subsets represent larger domain shifts.

Table 2 shows that the uncertainty benefits of *LENS* persist under distribution shift. Accuracy remains broadly comparable across methods on the smaller shifts, and all methods naturally degrade on the larger MMLU shifts; *LENS* remains competitive and obtains the highest accuracy on ARC-C with LENS_5 and on Physics with LENS_{10}. The calibration and likelihood results show a stronger pattern. Particularly *LENS* provides the strongest ECE across all tasks under distribution shift, improving from 8.63 to 3.98 on ARC-C and from 36.24 to 23.31 on Physics task. Moreover it improves NLL substantially across tasks, providing the strongest scores under smaller distribution shift and second best scores under larger shifts. These improvements indicate that *LENS* collected samples provide a useful uncertainty signal when the test distribution moves away from the fine-tuning distribution. SWAG remains close to deterministic LoRA in both ECE and NLL, suggesting that its local Gaussian perturbations do not expose enough predictive variation in this setting. TFB improves over deterministic and SWAG baselines on several shifted datasets, particularly ARC-E. Nevertheless, the reported *LENS* configurations achieve lower ECE and NLL on all four OOD datasets, while AS-10 remains strongest for NLL on Chemistry and Physics. This supports the view that following nonlinear loss-level geometry can yield better calibrated predictions under distribution shift than fixed local or linear approximations.

LENS can also be extended by aggregating level-set samples across independently trained models, further improving overall performance. Results on image classification and question-answering tasks are provided in Table 5, 4, and 6 of the Appendix J.

5 Discussion and Limitations

We introduced *LENS*, a *post-hoc* geometry-aware approach for uncertainty quantification based on exploring loss level sets around a trained model. The central idea is to construct an ensemble of predictors that remain close to the reference loss level while exposing predictive variation along directions not captured by a single point estimate. Across synthetic classification, UCI regression, CIFAR-10 image classification, LoRA-adapted language-model benchmarks, and distribution shift evaluations, *LENS* yields competitive predictive performance and consistently improves uncertainty quality, especially calibration and likelihood-based metrics. The results suggest that the proposed loss-preserving parameter variation is a useful source of uncertainty information. Unlike independently trained ensembles, *LENS* does not require training multiple independent models from scratch. Unlike local Gaussian or fixed-subspace approximations, it directly uses the geometry of the empirical loss landscape to define plausible alternatives to the trained solution. This makes the method especially appealing in settings where multiple independently trained models are expensive, such as parameter-efficient adaptation of foundation models.

At the same time, *LENS* has several limitations. First, although it is *post-hoc* and avoids full retraining, it still requires repeated loss and gradient evaluations during the construction of ensemble.This can be computationally expensive for large models, especially when many configurations or correction steps are required. Improving the efficiency, reducing the number of correction steps, and reusing gradient or curvature information are important directions for future work. Second, the current implementation relies on manually selected hyperparameters, including the level-set tolerance, tangent step size, and correction depth. In this work, we do not provide an automatic selection rule. Developing adaptive strategies for choosing these hyperparameters, possibly based on acceptance rate, loss residuals, or validation calibration, is an important practical extension. Finally, future work could study stronger theoretical guarantees, improved exploration of disconnected or highly curved level-set regions, and extensions beyond parameter-efficient fine-tuning to larger trainable parameter subsets.

AI USE DISCLOSURE

In this work, we used generative AI tools to assist with code editing and the implementation of an illustrative experiment, check the presentation and interpretation of experimental results, polish the manuscript for clarity, and improve the organization and formatting of tables and figures. All AI-assisted outputs were reviewed by the authors, and numerical claims were verified against the original experimental outputs. The authors take full responsibility for the final content of this work.

REFERENCES

Moloud Abdar, Farhad Pourpanah, Sadiq Hussain, Dana Rezazadegan, Li Liu, Mohammad Ghavamzadeh, Paul Fieguth, Xiaochun Cao, Abbas Khosravi, U. Rajendra Acharya, Vladimir Makarenkov, and Saeid Nahavandi. A review of uncertainty quantification in deep learning: Techniques, applications and challenges. *Information Fusion*, 76:243–297, 2021. ISSN 1566-2535. doi: https://doi.org/10.1016/j.inffus.2021.05.008. URL https://www.sciencedirect.com/science/article/pii/S1566253521001081.

ANM Nafiz Abeer, Sanket Jantre, Nathan M Urban, and Byung-Jun Yoon. Leveraging active subspaces to capture epistemic model uncertainty in deep generative models for molecular design. In *The 34th International Workshop on Machine Learning for Signal Processing (MLSP)*, London, UK, September 22-25 2024. IEEE.

ANM Nafiz Abeer, Sanket Jantre, Nathan M Urban, and Byung-Jun Yoon. Enhancing generative molecular design via uncertainty-guided fine-tuning of variational autoencoders. *Molecular Systems Design & Engineering*, 2026.

Maxwell Adam, Zach Furman, and Jesse Hoogland. The loss kernel: A geometric probe for deep learning interpretability, 2025. URL https://arxiv.org/abs/2509.26537.

Matan Atzmon, Niv Haim, Lior Yariv, Ofer Israelov, Haggai Maron, and Yaron Lipman. Controlling neural level sets. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/b20bb95ab626d93fd976af958fbc61ba-Paper.pdf.

Edmon Begoli, Tanmoy Bhattacharya, and Dimitri Kusnezov. The need for uncertainty quantification in machine-assisted medical decision making. *Nat. Mach. Intell.*, 1(1):20–23, January 2019.

Txus Blasco, J. Salvador Sánchez, and Vicente García. A survey on uncertainty quantification in deep learning for financial time series prediction. *Neurocomputing*, 576:127339, 2024. ISSN 0925-2312. doi: https://doi.org/10.1016/j.neucom.2024.127339. URL https://www.sciencedirect.com/science/article/pii/S0925231224001103.

David M. Blei, Alp Kucukelbir, and Jon D. McAuliffe. Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112(518):859–877, April 2017. ISSN 1537-274X. doi: 10.1080/01621459.2017.1285773. URL http://dx.doi.org/10.1080/01621459.2017.1285773.

Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural network. In *International conference on machine learning*, pp. 1613–1622. PMLR, 2015.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. Boolq: Exploring the surprising difficulty of natural yes/no questions. In *NAACL*, 2019.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge, 2018. URL https://arxiv.org/abs/1803.05457.

Paul G. Constantine, Eric Dow, and Qiqi Wang. Active subspace methods in theory and practice: Applications to kriging surfaces. *SIAM Journal on Scientific Computing*, 36(4):A1500–A1524, 2014. doi: 10.1137/130916138. URL https://doi.org/10.1137/130916138.

Paul G. Constantine, Carson Kent, and Tan Bui-Thanh. Accelerating markov chain monte carlo with active subspaces. *SIAM Journal on Scientific Computing*, 38(5):A2779–A2805, 2016. doi: 10.1137/15M1042127. URL https://doi.org/10.1137/15M1042127.

Paulo Cortez, A. Cerdeira, F. Almeida, T. Matos, and J. Reis. Wine quality. UCI Machine Learning Repository, 2009. DOI: 10.24432/C56S3T.

C. Cui, K. Zhang, T. Daulbaev, J. Gusak, I. Oseledets, and Z. Zhang. Active subspace of neural networks: Structural analysis and universal attacks. *SIAM Journal on Mathematics of Data Science*, 2(4), 2020.

E. Daxberger, A. Kristiadi, A. Immer*, R. Eschenhagen*, M. Bauer, and P. Hennig. Laplace redux — effortless bayesian deep learning. In *Advances in Neural Information Processing Systems 34 (NeurIPS 2021)*, pp. 20089–20103. Curran Associates, Inc., December 2021. URL https://proceedings.neurips.cc/paper/2021/file/a7c9585703d275249f30a088cebba0ad-Paper.pdf.

Daniel Dold, Emanuel Sommer, Julius Kobialka, Oliver Dürr, and David Rügamer. On the construction and implications of low-loss valleys in lora-based bayesian inference, 2026. URL https://arxiv.org/abs/2605.29580.

Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In Maria Florina Balcan and Kilian Q. Weinberger (eds.), *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pp. 1050–1059, New York, New York, USA, 20–22 Jun 2016. PMLR. URL https://proceedings.mlr.press/v48/gal16.html.

Timur Garipov, Pavel Izmailov, Dmitrii Podoprikhin, Dmitry Vetrov, and Andrew G Wilson. Loss surfaces, mode connectivity, and fast ensembling of dnns. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/be3087e74e9100d4bc4c6268cdbe8456-Paper.pdf.

Jakob Gawlikowski, Cedrique Rovile Njieutcheu Tassi, Mohsin Ali, Jongseok Lee, Matthias Humt, Jianxiang Feng, Anna Kruspe, Rudolph Triebel, Peter Jung, Ribana Roscher, Muhammad Shahzad, Wen Yang, Richard Bamler, and Xiao Xiang Zhu. A survey of uncertainty in deep neural networks. *Artif. Intell. Rev.*, 56(S1):1513–1589, October 2023.

J. Gerritsma, R. Onnink, and A. Versluis. Yacht hydrodynamics. UCI Machine Learning Repository, 1981. DOI: 10.24432/C5XG7R.

Ethan Goan and Clinton Fookes. *Bayesian Neural Networks: An Introduction and Survey*, pp. 45–87. Springer International Publishing, 2020. ISBN 9783030425531. doi: 10.1007/978-3-030-42553-1_3. URL http://dx.doi.org/10.1007/978-3-030-42553-1_3.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *Proceedings of the International Conference on Learning Representations*, 2019.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Eyke Hüllermeier and Willem Waegeman. Aleatoric and epistemic uncertainty in machine learning: an introduction to concepts and methods. *Mach. Learn.*, 110(3):457–506, March 2021.

Pavel Izmailov, Sharad Vikram, Matthew D Hoffman, and Andrew Gordon Wilson. What are bayesian neural network posteriors really like? In *International conference on machine learning*, pp. 4629–4640. PMLR, 2021.

Sanket Jantre, Nathan M. Urban, Xiaoning Qian, and Byung-Jun Yoon. Learning active subspaces for effective and scalable uncertainty quantification in deep neural networks. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2024.

Shane Jensen and Dean Foster. A Level-set Hit-and-run Sampler for Quasi-Concave Distributions. In Samuel Kaski and Jukka Corander (eds.), *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics*, volume 33 of *Proceedings of Machine Learning Research*, pp. 439–447, Reykjavik, Iceland, 22–25 Apr 2014. PMLR. URL https://proceedings.mlr.press/v33/jensen14.html.

Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/2650d6089a6d640c5e85b2b88265dc2b-Paper.pdf.

Benjamin Kompa, Jasper Snoek, and Andrew L Beam. Second opinion needed: communicating uncertainty in medical machine learning. *NPJ Digit. Med.*, 4(1):4, January 2021.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, 2009.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/9ef2ed4b7fd2c810847ffa5fa85bce38-Paper.pdf.

Wesley J Maddox, Pavel Izmailov, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. A simple baseline for bayesian uncertainty in deep learning. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/118921efba23fc329e6560b27861f0c2-Paper.pdf.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. Can a suit of armor conduct electricity? a new dataset for open book question answering, 2018. URL https://arxiv.org/abs/1809.02789.

Bálint Mucsányi, Michael Kirchhof, and Seong Joon Oh. Benchmarking uncertainty disentanglement: Specialized uncertainties for specialized tasks. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 50972–51038. Curran Associates, Inc., 2024. doi: 10.52202/079017-1614. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/5afa9cb1e917b898ad418216dc726fbd-Paper-Datasets_and_Benchmarks_Track.pdf.

Jishnu Mukhoti and Yarin Gal. Evaluating bayesian deep learning methods for semantic segmentation, 2019. URL https://arxiv.org/abs/1811.12709.

Qwen, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025. URL https://arxiv.org/abs/2412.15115.

Amir Hossein Rahmati, Sanket Jantre, Weifeng Zhang, Yucheng Wang, Byung-Jun Yoon, Nathan Urban, and Xiaoning Qian. C-loRA: Contextual low-rank adaptation for uncertainty estimation in large language models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=siPeAstQLq.

Trent Michael Russi. *Uncertainty Quantification with Experimental Data and Complex System Models*. PhD thesis, University of California, Berkeley, 2010. URL https://escholarship.org/uc/item/5d01s25c.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Winogrande: an adversarial winograd schema challenge at scale. *Commun. ACM*, 64(9):99–106, August 2021. ISSN 0001-0782. doi: 10.1145/3474381. URL https://doi.org/10.1145/3474381.

Haizhou Shi, Yibin Wang, Ligong Han, Huan Zhang, and Hao Wang. Training-free bayesianization for low-rank adapters of large language models. *arXiv preprint arXiv:2412.05723*, 2024.

Kamaljot Singh. Facebook Comment Volume. UCI Machine Learning Repository, 2015. DOI: https://doi.org/10.24432/C5Q886.

Athanasios Tsanas and Angeliki Xifara. Energy Efficiency. UCI Machine Learning Repository, 2012. DOI: https://doi.org/10.24432/C51307.

Binxu Wang and Carlos R. Ponce. On the level sets and invariance of neural tuning landscapes. In Sophia Sanborn, Christian Shewmake, Simone Azeglio, Arianna Di Bernardo, and Nina Miolane (eds.), *Proceedings of the 1st NeurIPS Workshop on Symmetry and Geometry in Neural Representations*, volume 197 of *Proceedings of Machine Learning Research*, pp. 278–300. PMLR, 03 Dec 2023. URL https://proceedings.mlr.press/v197/wang23a.html.

Yibin Wang, Haizhou Shi, Ligong Han, Dimitris N. Metaxas, and Hao Wang. BLob: Bayesian low-rank adaptation by backpropagation for large language models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=MaDykgj4Ru.

Zhuxuanzi Wang, Mingqiao Mo, Xi Xiao, Chen Liu, Chenrui Ma, Yunbei Zhang, Xiao Wang, Smita Krishnaswamy, and Tianyang Wang. Ctr-lora: Curvature-aware and trust-region guided low-rank adaptation for large language models. In *ICASSP 2026 - 2026 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1396–1400, 2026. doi: 10.1109/ICASSP55912.2026.11463994.

Max Welling and Yee Whye Teh. Bayesian learning via stochastic gradient langevin dynamics. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, ICML'11, pp. 681–688, Madison, WI, USA, 2011. Omnipress. ISBN 9781450306195.

Adam X. Yang, Maxime Robeyns, Xi Wang, and Laurence Aitchison. Bayesian low-rank adaptation for large language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=FJiUyzOF1m.

Olivier Zahm, Tiangang Cui, Kody Law, Alessio Spantini, and Youssef Marzouk. Certified dimension reduction in nonlinear bayesian inverse problems, 2022. URL https://arxiv.org/abs/1807.03712.

Ruqi Zhang, Chunyuan Li, Jianyi Zhang, Changyou Chen, and Andrew Gordon Wilson. Cyclical stochastic gradient mcmc for bayesian deep learning. *International Conference on Learning Representations*, 2020.

Bo Zhao, Nima Dehmamy, Robin Walters, and Rose Yu. Symmetry teleportation for accelerated optimization. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 16679–16690. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/69f7750aa28f75fddf101da038f8b529-Paper-Conference.pdf.

Qingping Zhou, Tengchao Yu, Xiaoqun Zhang, and Jinglai Li. Bayesian inference and uncertainty quantification for medical image reconstruction with poisson data. *SIAM Journal on Imaging Sciences*, 13(1):29–52, 2020. doi: 10.1137/19M1248352. URL https://doi.org/10.1137/19M1248352.

A Algorithmic Details

Algorithms 1, 2, and 3 provide the pseudo-code of each *LENS* implementation component.

Algorithm 1 Tangent Projection

Require: Direction d, gradient g, tolerance τ_g
if $\|g\|_2 \le \tau_g$ **then**
  return $d/\|d\|_2$
end if
$\widetilde{v} \leftarrow d - \dfrac{d^\top g}{\|g\|_2^2} g$
if $\|\widetilde{v}\|_2 \le \tau_g$ **then**
  return REJECT
end if
return $\widetilde{v}/\|\widetilde{v}\|_2$

Algorithm 2 Correction to the Target Level

Require: Initial proposal θ, target loss $\ell_\star$
Require: Tolerance ε, maximum steps M
Require: Damping η, step cap $s_{\max}$, gradient tolerance τ_g
for $m = 1, \dots, M$ **do**
  $\ell \leftarrow L(\theta)$
  $g \leftarrow \nabla L(\theta)$
  $r \leftarrow (\ell - \ell_\star)/\ell_\star$
  if $|r| \le \varepsilon$ **then**
    return $(\theta, \ell, g, \text{ACCEPT})$
  end if
  if $\|g\|_2^2 \le \tau_g$ **then**
    break
  end if
  $\Delta\theta \leftarrow -\eta \dfrac{r}{\|g\|_2^2} g$
  if $\|\Delta\theta\|_2 > s_{\max}$ **then**
    $\Delta\theta \leftarrow s_{\max}\Delta\theta/\|\Delta\theta\|_2$
  end if
  $\theta \leftarrow \theta + \Delta\theta$
end for
return $(\theta, L(\theta), \nabla L(\theta), \text{REJECT})$

Algorithm 3 is designed to construct a finite feasible ensemble; it does not require convergence to an invariant distribution or exhaustive coverage of the underlying level-set component.

B Mathematics of SWAG and Active Subspace

Let $\theta \in \mathbb{R}^p$ denote the trainable parameter vector sampled by each baseline.

SWAG-style sampling. Starting from $\theta_\star$, we run a fixed-learning-rate refinement and collect snapshots $\theta_1, \dots, \theta_T$. Let

$$\bar{\theta} = \frac{1}{T}\sum_{t=1}^{T} \theta_t, \qquad \widetilde{\Theta}_{t:} = (\theta_t - \bar{\theta})^\top.$$

SWAG uses the diagonal-plus-low-rank covariance

$$\widehat{\Sigma}_{\text{SWAG}} = \frac{1}{2}\left[\operatorname{diag}\left(\frac{1}{T}\sum_{t=1}^{T}(\theta_t - \bar{\theta})^2\right) + \frac{1}{T-1}\widetilde{\Theta}^\top\widetilde{\Theta}\right].$$

Algorithm 3 Level-Set Ensemble

Require: Loss L, trained parameters $\theta_\star$, number of samples N
Require: ε, $[\alpha_{\min}, \alpha_{\max}]$, σ_d
Require: Correction parameters M, η, $s_{\max}$, τ_g
$\ell_\star \leftarrow L(\theta_\star)$
$g_\star \leftarrow \nabla L(\theta_\star)$
Draw $d_0 \sim \mathcal{N}(0, I)$
$v_0 \leftarrow$ PROJECTTANGENT$(d_0, g_\star, \tau_g)$
if $v_0 =$ REJECT **then**
    $v_0 \leftarrow d_0 / \|d_0\|_2$
end if
Initialize two chains:

$$\theta^{(0)} = \theta_\star, \quad v^{(0)} = v_0, \qquad \theta^{(1)} = \theta_\star, \quad v^{(1)} = -v_0.$$

$g^{(0)} \leftarrow g_\star$, $g^{(1)} \leftarrow g_\star$
$\mathcal{S} \leftarrow \emptyset$
while $|\mathcal{S}| < N$ and budget remains **do**
    Select active chain $j \in \{0, 1\}$
    Draw $\xi \sim \mathcal{N}(0, I)$
    $\widehat{v} \leftarrow v^{(j)} + \sigma_d \xi$
    $v \leftarrow$ PROJECTTANGENT$(\widehat{v}, g^{(j)}, \tau_g)$
    if $v =$ REJECT **then**
        continue
    end if
    if $v^\top v^{(j)} < 0$ **then**
        $v \leftarrow -v$
    end if
    Draw $\alpha \sim \mathrm{Unif}(\alpha_{\min}, \alpha_{\max})$
    $\theta_{\mathrm{pred}} \leftarrow \theta^{(j)} + \alpha v$
    $(\theta_{\mathrm{new}}, \ell_{\mathrm{new}}, g_{\mathrm{new}}, a)$
    $\leftarrow$ CORRECTTOLEVEL$(\theta_{\mathrm{pred}}, \ell_\star, \varepsilon, M, \eta, s_{\max}, \tau_g)$
    if $a =$ ACCEPT **then**
        $\mathcal{S} \leftarrow \mathcal{S} \cup \{\theta_{\mathrm{new}}\}$
        $\theta^{(j)} \leftarrow \theta_{\mathrm{new}}$
        $g^{(j)} \leftarrow g_{\mathrm{new}}$
        $v^{(j)} \leftarrow v$
    end if
end while
return $\mathcal{S}$

In practice, samples are drawn without forming this covariance:

$$\theta^{(s)} = \bar{\theta} + \frac{\gamma}{\sqrt{2}} \left(\sigma_{\mathrm{diag}} \odot z_D + \frac{z_L^\top \widetilde{\Theta}}{\sqrt{T-1}} \right),$$

where $z_D \sim \mathcal{N}(0, I_p)$, $z_L \sim \mathcal{N}(0, I_T)$, and γ is the sampling scale. These samples summarize local variability along the refinement trajectory and are not constrained to preserve $L(\theta_\star)$.

Active-subspace sampling. For mini-batches $\mathcal{B}_m$, we collect gradients

$$g_m = \nabla_\theta L_{\mathcal{B}_m}(\theta_\star), \qquad m = 1, \ldots, M,$$

and form the empirical gradient second moment

$$C = \frac{1}{M} \sum_{m=1}^{M} g_m g_m^\top.$$

Let $v_1, \ldots, v_K$ be the top eigenvectors with eigenvalues $\lambda_1, \ldots, \lambda_K$. In high dimensions, these directions are computed through the Gram matrix $GG^\top/M$, where G has rows $g_m^\top$. With $P =$

$[v_1^\top; \ldots; v_K^\top]$, samples are drawn as

$$\theta^{(s)} = \theta_\star + P^\top z^{(s)}, \qquad z_k^{(s)} \sim \mathcal{N}(0, \nu_k)\,,$$

where,

$$\nu_k = \frac{\sigma_z^2}{\sigma_0^{-2} + c\lambda_k}$$

Here, σ_0 and σ_z control the sampling scale, while $c > 0$ accounts for the task-specific gradient and data scaling. Hence, directions with larger eigenvalues receive smaller perturbations. The finite set of reduced coordinates are centered around $\theta_\star$. Active-subspace sampling is therefore a local linear reduced-space approximation rather than a sampler constrained to nonlinear loss-level geometry.

C Curvature and the Magnitude of the Normal Corrector

This section relates the size of the normal correction to the directional curvature of the loss level set. Let

$$F(\theta) = L(\theta) - \ell_\star, \qquad g_0 = \nabla L(\theta_0), \qquad H_0 = \nabla^2 L(\theta_0),$$

where θ_0 is an accepted point satisfying $|F(\theta_0)| \le \varepsilon\ell_\star$ and $\|g_0\|_2 \neq 0$. Let v be a unit tangent direction at θ_0, so that $g_0^\top v = 0$, and define the predictor point

$$\theta_{\mathrm{p}} = \theta_0 + \alpha v.$$

Assuming L is twice continuously differentiable near θ_0, Taylor expansion gives

$$\begin{aligned} F(\theta_{\mathrm{p}}) &= F(\theta_0) + \alpha g_0^\top v + \frac{\alpha^2}{2} v^\top H_0 v + o(\alpha^2) \\ &= F(\theta_0) + \frac{\alpha^2}{2} v^\top H_0 v + o(\alpha^2). \end{aligned}$$

Thus, even though the predictor is tangent to the level set to first order, its finite-step loss drift contains the directional quadratic form $v^\top H_0 v$.

Let $g_{\mathrm{p}} = \nabla L(\theta_{\mathrm{p}})$. Before clipping, our corrector is

$$\Delta\theta_{\mathrm{unc}} = -\eta \frac{r(\theta_{\mathrm{p}})}{\|g_{\mathrm{p}}\|_2^2} g_{\mathrm{p}}, \qquad r(\theta_{\mathrm{p}}) = \frac{F(\theta_{\mathrm{p}})}{\ell_\star}.$$

Its norm is therefore

$$\begin{aligned} \|\Delta\theta_{\mathrm{unc}}\|_2 &= \frac{\eta}{\ell_\star \|g_{\mathrm{p}}\|_2} \left| F(\theta_0) + \frac{\alpha^2}{2} v^\top H_0 v + o(\alpha^2) \right| \\ &\le \frac{\eta\varepsilon}{\|g_{\mathrm{p}}\|_2} + \frac{\eta\alpha^2}{2\ell_\star} \frac{|v^\top H_0 v|}{\|g_{\mathrm{p}}\|_2} + o(\alpha^2). \end{aligned}$$

The directional normal curvature of the level set, up to the choice of sign convention, is

$$\kappa_n(\theta_0; v) = \frac{v^\top H_0 v}{\|g_0\|_2}.$$

Since $g_{\mathrm{p}} = g_0 + O(\alpha)$, the curvature-dependent term in the bound satisfies

$$\frac{\eta\alpha^2}{2\ell_\star} \frac{|v^\top H_0 v|}{\|g_{\mathrm{p}}\|_2} = \frac{\eta\alpha^2}{2\ell_\star} |\kappa_n(\theta_0; v)| + O(\alpha^3).$$

Hence, up to the bounded residual contribution $\eta\varepsilon/\|g_{\mathrm{p}}\|_2$ and local gradient scaling, the curvature-dependent contribution to the required uncapped normal correction grows with the directional normal curvature.

The implemented update clips this quantity:

$$\|\Delta\theta\|_2 = \min\{\|\Delta\theta_{\mathrm{unc}}\|_2,\, s_{\max}\}.$$

Therefore, the curvature relation describes the required *uncapped* correction and the likelihood that clipping is activated. Once clipping is active, the applied correction norm equals $s_{\max}$.

D CORRECTOR ACCEPTANCE AND ENSEMBLE COMPLETION

Across the QA runs, *LENS* retained 99.7% of the requested configurations. Every run reached its requested ensemble size except one WG-M run, which retained 18 of 20 configurations. Every retained member independently satisfied the prescribed relative-loss tolerance, while proposals that failed correction were excluded. Thus, the proposal budget was sufficient to construct nearly all requested ensembles while enforcing the loss-tube constraint.

E COMPUTATIONAL COST OF LENS

Let p denote the number of parameters explored by *LENS*, T the number of attempted proposals, and M the maximum number of correction iterations per proposal. Let C_{fg} denote the cost of one combined loss-and-gradient evaluation on the data used to construct the level set. The original model-training cost is excluded because *LENS* is applied post hoc to an already-trained model.

At initialization, *LENS* evaluates the target loss $\ell_\star = L(\theta_\star)$ and its gradient. We denote this one-time cost by C_{init}, which is approximately $C_{\mathrm{fg}} + O(p)$. For each subsequent proposal, constructing and projecting a direction onto the tangent space requires vector inner products, norms, and vector updates, each linear in the number of sampled parameters. Because the gradient at the current accepted point is cached, the tangent projection does not require an additional backward pass and has cost $O(p)$.

Let $m_t \le M$ denote the number of correction iterations used for proposal t. Each correction iteration requires one loss-and-gradient evaluation, with cost C_{fg}, together with $O(p)$ parameter-vector operations for computing the gradient norm, constructing $\Delta\theta$, clipping its norm, and updating the parameters. The total post-hoc sampling cost is therefore

$$C_{\mathrm{LENS}} = C_{\mathrm{init}} + \sum_{t=1}^{T} \big[O(p) + m_t\big(C_{\mathrm{fg}} + O(p)\big)\big] .$$

Using the maximum correction budget gives the upper bound

$$C_{\mathrm{LENS}} = O(TMC_{\mathrm{fg}} + T(M+1)p) ,$$

apart from the one-time initialization cost. In practice, if

$$\overline{m} = \frac{1}{T}\sum_{t=1}^{T} m_t$$

is the average number of correction iterations, a more representative expression is

$$C_{\mathrm{LENS}} \approx C_{\mathrm{init}} + T\left[O(p) + \overline{m}\big(C_{\mathrm{fg}} + O(p)\big)\right] .$$

If N accepted samples are required and the empirical acceptance rate is ρ, then $T \approx N/\rho$. Consequently,

$$C_{\mathrm{LENS}} \approx C_{\mathrm{init}} + \frac{N}{\rho}\left[O(p) + \overline{m}\big(C_{\mathrm{fg}} + O(p)\big)\right] .$$

To interpret C_{fg}, suppose the construction dataset contains n_c examples and uses batches of size B. One loss-and-gradient evaluation processes

$$q = \left\lceil \frac{n_c}{B} \right\rceil$$

mini-batches, giving

$$C_{\mathrm{fg}} \approx q\left(C_{\mathrm{forward}}(B) + C_{\mathrm{backward}}(B)\right) .$$

Thus, when each correction evaluates the full construction dataset, one correction iteration is comparable to one training epoch over that dataset, excluding the optimizer update. The repeated loss-and-gradient evaluations generally dominate the $O(p)$ parameter-vector operations.

With respect to parameter storage, *LENS* can be implemented dynamically by generating one sample at a time and immediately accumulating its predictions. This requires only a constant number

of parameter, gradient, and direction vectors and therefore has $O(p)$ parameter memory, the same asymptotic order as deterministic inference, although with a larger constant factor and the usual activation memory required for backpropagation. If all accepted parameter samples are stored explicitly, the memory cost is $O(Np)$.

To empirically show the overhead caused by repeated loss and gradient evaluations, we measure post-hoc construction time on Facebook and Wine-red (Figure 4). These measurements compare the end-to-end post-hoc construction costs of the configurations used in our main experiments. Consistent with the analysis, *LENS* is slower than SWAG and AS because each proposal may require multiple correction iterations, although its absolute construction time remains below 0.5 seconds for these small models.

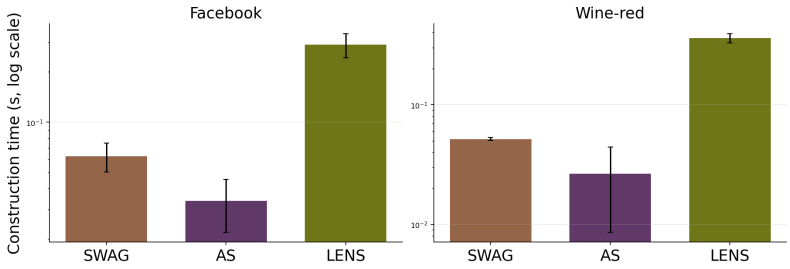


Figure 4: Post-hoc construction time for 20 samples, reported as mean $\pm$ standard deviation over three seeds. Base-model training, checkpoint loading, and test-time prediction are excluded. The vertical axis uses a logarithmic scale.

F METRICS

For classification, let $\widehat{p}_i(c)$ be the predictive probability assigned to class c on example i, and let y_i be the true label. For an ensemble or sampled method,

$$\widehat{p}_i(c) = \frac{1}{S}\sum_{s=1}^{S} p_{\theta^{(s)}}(c \mid x_i).$$

The predicted label and confidence are

$$\widehat{y}_i = \arg\max_c \widehat{p}_i(c), \qquad \widehat{c}_i = \max_c \widehat{p}_i(c).$$

F.1 EXPECTED CALIBRATION ERROR

Expected calibration error (ECE) measures the gap between empirical accuracy and predicted confidence. Partition the interval $[0, 1]$ into M bins $\{I_m\}_{m=1}^{M}$, and define

$$B_m = \{i : \widehat{c}_i \in I_m\}.$$

The empirical accuracy and average confidence in bin m are

$$\mathrm{acc}(B_m) = \frac{1}{|B_m|}\sum_{i\in B_m} \mathbf{1}\{\widehat{y}_i = y_i\},$$

and

$$\mathrm{conf}(B_m) = \frac{1}{|B_m|}\sum_{i\in B_m} \widehat{c}_i.$$

The ECE is

$$\mathrm{ECE} = \sum_{m=1}^{M} \frac{|B_m|}{n} \left|\mathrm{acc}(B_m) - \mathrm{conf}(B_m)\right|.$$

Lower ECE indicates that confidence better matches empirical correctness. In the paper, ECE is reported as a percentage.

F.2 Negative Log-likelihood

Negative log-likelihood (NLL) evaluates the probability assigned to the observed label:

$$\mathrm{NLL}_{\mathrm{cls}} = -\frac{1}{n}\sum_{i=1}^{n} \log \widehat{p}_i(y_i).$$

It penalizes confident mistakes strongly, so it is sensitive to both accuracy and calibration. For regression experiments with a Gaussian predictive distribution

$$y_i \mid x_i \sim \mathcal{N}(\widehat{\mu}_i, \widehat{\sigma}_i^2),$$

the NLL is

$$\mathrm{NLL}_{\mathrm{reg}} = \frac{1}{n}\sum_{i=1}^{n}\left[\frac{1}{2}\log(2\pi\widehat{\sigma}_i^2) + \frac{(y_i-\widehat{\mu}_i)^2}{2\widehat{\sigma}_i^2}\right].$$

Thus regression NLL rewards accurate means and calibrated predictive variances.

F.3 Brier Score

The Brier score measures the mean squared error between the predicted class probabilities and the observed outcome. It evaluates the full predictive distribution rather than only the most likely class, so it penalizes both incorrect predictions and poorly calibrated confidence. Lower Brier scores indicate better probabilistic predictions.

$$\mathrm{Brier} = \frac{1}{n}\sum_{i=1}^{n}\sum_{c}\left(\widehat{p}_i(c) - \mathbf{1}\{y_i = c\}\right)^2.$$

G Aleatoric and Epistemic Uncertainty

Uncertainty is often decomposed into aleatoric and epistemic components (Kendall & Gal, 2017; Hüllermeier & Waegeman, 2021). Aleatoric uncertainty reflects ambiguity that remains even if the model parameters are known, such as label noise or class overlap. Epistemic uncertainty reflects uncertainty about the model itself, and can decrease when more informative data or stronger constraints are available.

Let $q(\theta)$ denote the distribution over sampled parameters. For *LENS*, q is the empirical uniform distribution over accepted level-set samples. The averaged predictive distribution is

$$\bar{p}(y \mid x) = \mathbb{E}_{\theta\sim q}\left[p_\theta(y \mid x)\right] \approx \frac{1}{S}\sum_{s=1}^{S} p_{\theta^{(s)}}(y \mid x).$$

The total classification uncertainty is the predictive entropy

$$\mathcal{U}_{\mathrm{total}}(x) = H[\bar{p}(\cdot \mid x)] = -\sum_{y}\bar{p}(y \mid x)\log\bar{p}(y \mid x).$$

The aleatoric component is estimated by the expected entropy of individual predictors,

$$\mathcal{U}_{\mathrm{alea}}(x) = \mathbb{E}_{\theta\sim q}\left[H[p_\theta(\cdot \mid x)]\right] \approx \frac{1}{S}\sum_{s=1}^{S} H[p_{\theta^{(s)}}(\cdot \mid x)].$$

The epistemic component is the mutual information between the prediction and the sampled parameters,

$$\begin{aligned}\mathcal{U}_{\mathrm{epi}}(x) &= I(y, \theta \mid x, \mathcal{D}) \\ &= H[\bar{p}(\cdot \mid x)] - \mathbb{E}_{\theta\sim q}\left[H[p_\theta(\cdot \mid x)]\right].\end{aligned}$$

Equivalently, epistemic uncertainty is high when individual sampled models are confident but disagree with each other. This is the effect *LENS* is designed to expose: among models that preserve the loss, disagreement on a test input indicates that the objective does not uniquely determine that prediction.

	UCI regression	CIFAR-10	Foundation QA
Base train.	MLP-32, dropout 0.05; AdamW; batch 32; LR 10^{-3}; WD 10^{-4}	ResNet-20; SGD; 200 ep.; batch 128; LR 0.1; WD 5×10^{-4}	Qwen2.5-7B + 4-bit LoRA (r=8); batch 4; 5000 steps; LR 10^{-4}
Det. / Ens.	1 / seed; 3-seed ensemble	1 / seed; 3-seed ensemble	1 LoRA / seed; 3-seed ensemble
MCD	K=20	K=20	K=20
SWAG	K=20, T=200, Δ=20, η=10^{-3}, s=1.0	K=20, T=500, Δ=25, η=10^{-2}, s=1.0	$k\in\{5,10,20\}$, T=500, Δ=25, η=10^{-5}, s=0.5
AS	K=20, M=40, d=3, σ_0=0.1, σ_z=1.0	K=20, M=50, d=10, σ_0=0.05, σ_z=0.5	$K\in\{5,10,20\}$, M=100, $d\in\{3,10\}$, σ_0=0.1, σ_z=1.0
LENS	K=20, ε=$10^{-3\dagger}$, $\alpha_{\max}$=5, trials 5000, steps 30	K=20, ε=0.05, $\alpha_{\max}$=5, trials 50, steps 30	$K\in\{5,10,20\}$, ε=10^{-1}, $\alpha_{\max}$=30, trials 50, steps 20

Table 3: Training and UQ hyperparameters. Notation: K parameter samples, T SWAG refinement steps, Δ SWAG collection interval, M gradient batches, d subspace dimension. Seeds are $\{0,1,2\}$ for UCI/CIFAR-10 and $\{32,42,52\}$ for Foundation QA. †Energy uses ϵ=10^{-4}.

H Additional Toy-example Details

H.1 Two-dimensional Level-set Toy Example

The two-dimensional toy example isolates the sampling geometry in the parameter space. Let $\theta = (\theta_1, \theta_2)$ and use the predictor

$$\widehat{y}_\theta(x) = \theta_2 - \theta_1^2 + \sqrt{\mu}(\theta_1 - c)x.$$

With data points $(0,0)$ and $(1,0)$, the residuals are

$$r_0(\theta) = \theta_2 - \theta_1^2, \qquad r_1(\theta) = \theta_2 - \theta_1^2 + \sqrt{\mu}(\theta_1 - c),$$

and the loss is

$$L(\theta) = \frac{1}{2}\left[r_0(\theta)^2 + r_1(\theta)^2\right].$$

This loss has a curved low-loss region, roughly following $\theta_2 \approx \theta_1^2$. The exact zero-loss solution used in the notebook is

$$\theta_{\text{opt}} = (c, c^2),$$

and the plotted configuration uses $\mu = 0.35$ and $c = 0.81$. However, the comparison is *post-hoc*: the samplers are initialized from an early-stopped $\theta_\star$ rather than from the exact optimum. When $\nabla L(\theta_\star) \neq 0$, the implicit-function theorem gives a locally regular one-dimensional level set

$$\mathcal{M}_{\ell_\star} = \{\theta : L(\theta) = L(\theta_\star)\},$$

whose tangent vectors satisfy

$$\nabla L(\theta_\star)^\top v = 0.$$

In two dimensions, a unit tangent can be written explicitly as

$$v = \frac{1}{\|\nabla L(\theta)\|_2}\begin{bmatrix} -\partial L/\partial\theta_2 \\ \partial L/\partial\theta_1 \end{bmatrix}.$$

The level-set path alternates tangent predictor steps with normal correction steps, so its samples remain close to the contour $L(\theta) = L(\theta_\star)$. SWAG instead forms a local Gaussian from a refinement trajectory, and active-subspace sampling forms a reduced Gaussian around $\theta_\star$. These alternatives can produce useful nearby parameters, but neither is explicitly constrained to stay on the target contour.

The prediction-space panel evaluates the sampled predictors over a range of x values. All methods can preserve the observed points, but their extrapolations away from $x = 0$ and $x = 1$ differ. This illustrates why parameter-space geometry matters for uncertainty: even when sampled parameters have similar training behavior, they can imply different predictive variability away from observed data.

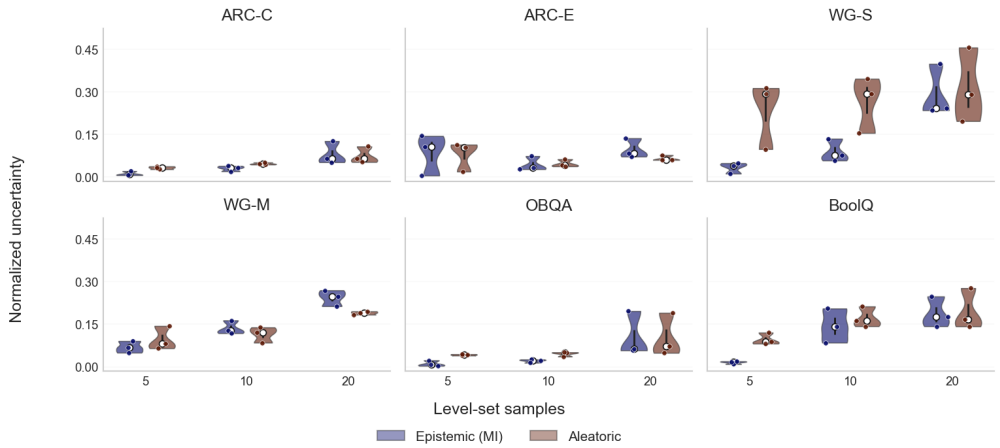


Figure 5: Epistemic and aleatoric uncertainty under *LENS* as the number of samples increases. Aleatoric uncertainty remains relatively stable for each dataset, whereas epistemic uncertainty generally increases with more level-set samples, indicating greater disagreement among comparable-loss models.

I Epistemic and Aleatoric Uncertainty

Beyond aggregate calibration metrics, we examine how level-set samples contribute to different sources of predictive uncertainty. Predictive entropy measures the total uncertainty of the average prediction, while mutual information captures disagreement among sampled models and is commonly used as an epistemic uncertainty measure in Bayesian and ensemble-based uncertainty estimation (Mukhoti & Gal, 2019; Mucsányi et al., 2024).

Figure 5 reports normalized epistemic and aleatoric uncertainty for different numbers of level-set samples. Across datasets, aleatoric uncertainty remains concentrated around a dataset-specific range, suggesting that the intrinsic ambiguity captured by individual predictors changes only mildly as more samples are added. In contrast, epistemic uncertainty generally increases with the number of level-set samples, indicating that additional samples expose more disagreement among comparable-loss models. This supports the role of *LENS* as a mechanism for revealing predictive instability along loss-preserving parameter directions.

J Additional Experiments

We additionally investigate *Ensemble-LENS*, which combines the *LENS* predictive distributions obtained from three independently trained models and therefore captures both variation across training runs and variation along each model's loss level set. The main results already show the benefit of single-model *LENS* over deterministic inference: on CIFAR-10-C, for example, *LENS* reduces NLL from 1.989 to 1.581, ECE from 26.57% to 16.00%, and Brier score from 0.705 to 0.635, while approximately preserving accuracy. In table 4 *Ensemble-LENS* improves these results further, achieving 55.75% accuracy, 0.9052 AUROC, 1.424 NLL, 10.40% ECE, and 0.591 Brier score. These values improve upon both the conventional deterministic ensemble, which obtains 55.53%, 0.9009, 1.692, 19.94%, and 0.645, respectively, and the best value reported in the main table for every metric. Relative to the best main-table results, *Ensemble-LENS* reduces NLL by 9.9%, ECE by 5.60 percentage points (35.0% relative), and Brier score by 6.9%, while also increasing accuracy by 0.22 percentage points and AUROC by 0.0043.

A similar pattern appears in QA. In Table 5, on the six in-distribution datasets, the best *Ensemble-LENS* configuration reduces NLL relative to the best previously reported result from 0.61 to 0.57 on WG-S, 0.40 to 0.34 on ARC-C, 0.15 to 0.13 on ARC-E, 0.54 to 0.47 on WG-M, 0.26 to 0.21 on OBQA, and 0.33 to 0.28 on BoolQ. It also improves the best reported ECE from 9.56% to 6.88% on WG-S, 3.97% to 1.66% on ARC-C, 8.47% to 5.49% on WG-M, 4.00% to 3.80% on OBQA, and 4.50% to 2.49% on BoolQ. On ARC-E, TFB retains the lowest ECE compared with *Ensemble-*

LENS. In Table 6, under distribution shift, *Ensemble-LENS* lowers NLL below the best main-table result on three datasets: from 0.26 to 0.21 on OBQA, 0.37 to 0.32 on ARC-C, and 0.16 to 0.14 on ARC-E. On Physics, the two results are tied at 1.35 at the reported precision. On Chemistry, AS_{10} retains the lowest NLL of 1.40, compared with 1.48 for *Ensemble-LENS*. The largest calibration gains occur on the stronger shifts, where ECE decreases from 23.89% to 22.40% on Chemistry and from 23.31% to 19.70% on Physics. Although the gains are not universal—the conventional ensemble remains stronger on in-distribution CIFAR-10—these results demonstrate that diversity across independently trained models and diversity within their loss level sets can be complementary. We therefore present *Ensemble-LENS* as a proof-of-concept extension rather than part of the primary single-model post-hoc setting.

Method	Acc. (%) ↑	AUROC ↑	NLL ↓	ECE (%) ↓	Brier ↓
In-distribution: CIFAR-10					
Ens.	**92.79**	**0.9965**	**0.218**	**0.67**	**0.107**
LENS	$91.00_{\pm 0.35}$	$0.9947_{\pm 0.0004}$	$0.271_{\pm 0.011}$	$1.22_{\pm 0.79}$	$0.132_{\pm 0.006}$
Ens.-LENS	92.53	0.9962	0.239	3.80	0.115
Out-of-distribution: CIFAR-10-C					
Ens.	55.53	0.9009	1.692	19.94	0.645
LENS	$53.60_{\pm 0.58}$	$0.8938_{\pm 0.0023}$	$1.581_{\pm 0.024}$	$16.00_{\pm 1.89}$	$0.635_{\pm 0.004}$
Ens.-LENS	**55.75**	**0.9052**	**1.424**	**10.40**	**0.591**

Table 4: Comparison of the independently trained three-model ensemble, standard single-model *LENS*, and *Ensemble-LENS* on CIFAR-10 and CIFAR-10-C. *Ensemble-LENS* averages the *LENS* predictive distributions from three independently trained models, corresponding to 60 *LENS* members. Standard *LENS* reports mean and standard deviation over three seeds; the two ensemble variants are evaluated as single aggregate predictors. Best and second-best values are bolded and underlined, respectively.

Metric	Method	WG-S	ARC-C	ARC-E	WG-M	OBQA	BoolQ
Acc. (%) ↑	Ens.	73.56	89.51	**96.84**	**80.82**	93.00	89.11
	$LENS_5$	$72.14_{\pm 2.95}$	$89.42_{\pm 0.23}$	$95.79_{\pm 0.55}$	$79.85_{\pm 0.56}$	$92.93_{\pm 1.10}$	$89.14_{\pm 0.12}$
	Ens.-$LENS_5$	**73.72**	89.68	96.51	80.43	93.20	**89.30**
	$LENS_{10}$	$72.03_{\pm 2.92}$	$89.39_{\pm 0.34}$	$96.44_{\pm 0.16}$	$80.06_{\pm 0.84}$	$93.00_{\pm 0.72}$	$89.07_{\pm 0.04}$
	Ens.-$LENS_{10}$	73.32	89.25	96.63	**80.82**	**93.40**	89.11
	$LENS_{20}$	$72.09_{\pm 1.89}$	$89.25_{\pm 0.37}$	$96.25_{\pm 0.15}$	$79.01_{\pm 0.39}$	$92.60_{\pm 1.40}$	$88.94_{\pm 0.34}$
	Ens.-$LENS_{20}$	73.09	**89.76**	96.51	80.03	93.20	89.14
ECE (%) ↓	Ens.	17.10	7.67	2.38	13.53	4.49	8.04
	$LENS_5$	$20.45_{\pm 2.70}$	$8.11_{\pm 0.29}$	$8.78_{\pm 5.78}$	$15.70_{\pm 1.04}$	$5.07_{\pm 1.00}$	$8.15_{\pm 0.49}$
	Ens.-$LENS_5$	15.75	6.71	8.67	11.87	4.11	7.08
	$LENS_{10}$	$18.06_{\pm 2.90}$	$6.64_{\pm 0.82}$	$\mathbf{1.98_{\pm 2.06}}$	$13.01_{\pm 1.20}$	$4.43_{\pm 0.49}$	$4.50_{\pm 0.67}$
	Ens.-$LENS_{10}$	13.69	5.71	2.18	10.12	**3.80**	**2.49**
	$LENS_{20}$	$9.76_{\pm 3.58}$	$3.97_{\pm 0.81}$	$4.89_{\pm 2.03}$	$8.47_{\pm 0.08}$	$5.81_{\pm 5.01}$	$4.63_{\pm 1.37}$
	Ens.-$LENS_{20}$	**6.88**	**1.66**	5.62	**5.49**	4.92	2.52
NLL ↓	Ens.	0.82	0.52	0.15	0.75	0.26	0.42
	$LENS_5$	$0.98_{\pm 0.22}$	$0.52_{\pm 0.03}$	$0.26_{\pm 0.09}$	$0.87_{\pm 0.09}$	$0.29_{\pm 0.04}$	$0.42_{\pm 0.03}$
	Ens.-$LENS_5$	0.75	0.46	0.20	0.62	0.24	0.36
	$LENS_{10}$	$0.88_{\pm 0.14}$	$0.44_{\pm 0.06}$	$0.16_{\pm 0.04}$	$0.70_{\pm 0.06}$	$0.26_{\pm 0.02}$	$0.33_{\pm 0.01}$
	Ens.-$LENS_{10}$	0.67	0.37	**0.13**	0.55	**0.21**	0.29
	$LENS_{20}$	$0.66_{\pm 0.09}$	$0.40_{\pm 0.03}$	$0.18_{\pm 0.03}$	$0.54_{\pm 0.02}$	$0.28_{\pm 0.07}$	$0.34_{\pm 0.03}$
	Ens.-$LENS_{20}$	**0.57**	**0.34**	0.16	**0.47**	0.24	**0.28**

Table 5: In-distribution QA comparison between the independently trained three-model ensemble, standard single-model *LENS*, and *Ensemble-LENS*. $LENS_K$ reports mean and standard deviation over three seeds, whereas Ens.-$LENS_K$ averages all accepted *LENS* members across those seeds and is evaluated as one predictive ensemble. *Ensemble-LENS* contains up to $3K$ accepted members.

K Experimental Settings

Implementation details for each task are summarized in Table 3. For the reported experiments, construction and final evaluation use disjoint data splits. *LENS* and AS are constructed using the validation split: *LENS* uses validation loss and gradients to define and correct towards the target level set, while AS uses validation mini-batch gradients to fit the gradient second-moment subspace. SWAG is constructed from a fixed-learning-rate refinement trajectory on the training split, starting from the trained checkpoint.

Hyperparameters for AS, SWAG, and *LENS* are selected using validation performance; after selection, final evaluation metrics are reported on held-out test or OOD splits. In particular, AS is tuned over subspace dimension and sampling scale, SWAG over refinement learning rate and sampling

Metric	Method	OBQA	ARC-C	ARC-E	Chem	Phy
Acc. (%) ↑	Ens.	93.00	89.08	95.66	**51.00**	48.04
	LENS_5	$92.93_{\pm 1.10}$	$89.33_{\pm 0.31}$	$95.64_{\pm 0.22}$	$\mathbf{51.00}_{\pm \mathbf{1.00}}$	$49.67_{\pm 3.15}$
	Ens.-LENS_5	93.20	89.16	**95.79**	**51.00**	48.04
	LENS_{10}	$93.00_{\pm 0.72}$	$89.28_{\pm 0.18}$	$95.57_{\pm 0.35}$	$50.00_{\pm 1.00}$	$50.33_{\pm 4.93}$
	Ens.-LENS_{10}	**93.40**	**89.51**	95.71	**51.00**	**51.96**
	LENS_{20}	$92.60_{\pm 1.40}$	$88.94_{\pm 0.47}$	$95.23_{\pm 0.27}$	$50.00_{\pm 3.61}$	$48.69_{\pm 3.15}$
	Ens.-LENS_{20}	93.20	89.08	95.50	**51.00**	50.00
ECE (%) ↓	Ens.	4.49	7.97	2.54	33.07	37.59
	LENS_5	$5.07_{\pm 1.00}$	$7.92_{\pm 0.10}$	$2.88_{\pm 0.20}$	$31.98_{\pm 0.14}$	$33.63_{\pm 2.95}$
	Ens.-LENS_5	4.11	7.24	2.24	31.56	33.20
	LENS_{10}	$4.43_{\pm 0.49}$	$6.46_{\pm 1.77}$	$1.91_{\pm 0.80}$	$32.07_{\pm 4.87}$	$30.16_{\pm 3.47}$
	Ens.-LENS_{10}	**3.80**	5.68	**1.21**	28.55	28.33
	LENS_{20}	$5.81_{\pm 5.01}$	$3.98_{\pm 1.02}$	$1.86_{\pm 1.02}$	$23.89_{\pm 2.57}$	$23.31_{\pm 6.52}$
	Ens.-LENS_{20}	4.92	**3.58**	2.84	**22.40**	**19.70**
NLL ↓	Ens.	0.26	0.42	0.16	1.83	1.63
	LENS_5	$0.29_{\pm 0.04}$	$0.44_{\pm 0.03}$	$0.17_{\pm 0.02}$	$1.88_{\pm 0.13}$	$1.63_{\pm 0.12}$
	Ens.-LENS_5	0.24	0.39	0.15	1.80	1.55
	LENS_{10}	$0.26_{\pm 0.02}$	$0.41_{\pm 0.03}$	$0.16_{\pm 0.01}$	$1.80_{\pm 0.06}$	$1.53_{\pm 0.05}$
	Ens.-LENS_{10}	**0.21**	0.35	**0.14**	1.71	1.41
	LENS_{20}	$0.28_{\pm 0.07}$	$0.37_{\pm 0.03}$	$0.17_{\pm 0.02}$	$1.61_{\pm 0.15}$	$1.44_{\pm 0.13}$
	Ens.-LENS_{20}	0.24	**0.32**	0.15	**1.48**	**1.35**

Table 6: QA comparison under distribution shift for models trained on OBQA. Standard *LENS* reports mean and standard deviation over three seeds; *Ensemble-LENS* averages all corresponding *LENS* members across the seeds and is evaluated once. OBQA is in-distribution, ARC-C and ARC-E are smaller shifts, and Chemistry and Physics are larger shifts.

Metric	Result	WG-S	ARC-C	ARC-E	WG-M	OBQA	BoolQ
Acc. (%) ↑	Main-table best	73.56	89.62	**96.84**	**80.82**	93.00	89.22
	Ens.-LENS best	**73.72**	**89.76**	96.63	**80.82**	**93.40**	**89.30**
ECE (%) ↓	Main-table best	9.56	3.97	**1.91**	8.47	4.00	4.50
	Ens.-LENS best	**6.88**	**1.66**	2.18	**5.49**	**3.80**	**2.49**
NLL ↓	Main-table best	0.61	0.40	0.15	0.54	0.26	0.33
	Ens.-LENS best	**0.57**	**0.34**	**0.13**	**0.47**	**0.21**	**0.28**

Table 7: In-distribution comparison between the best results in the main results tables and the best *Ensemble-LENS* results. Results are selected independently for each metric and dataset and may therefore correspond to different method configurations. Better values are bolded and the other values are underlined; ties are both bolded.

Metric	Result	OBQA	ARC-C	ARC-E	Chem	Phy
Acc. (%) ↑	Main-table best	93.00	89.33	95.66	**51.00**	50.33
	Ens.-LENS best	**93.40**	**89.51**	**95.79**	**51.00**	**51.96**
ECE (%) ↓	Main-table best	4.00	3.98	1.86	23.89	23.31
	Ens.-LENS best	**3.80**	**3.58**	**1.21**	**22.40**	**19.70**
NLL ↓	Main-table best	0.26	0.37	0.16	**1.40**	**1.35**
	Ens.-LENS best	**0.21**	**0.32**	**0.14**	1.48	**1.35**

Table 8: Comparison under distribution shift between the best results in the main results tables and the best *Ensemble-LENS* results. Results are selected independently for each metric and dataset and may therefore correspond to different method configurations. Better values are bolded and the other values are underlined; ties are both bolded.

scale, and *LENS* over step size and correction depth.[2] For TFB, we follow the implementation and hyperparameter-selection protocol from the authors' official repository. For each seed, β is selected from $[10^{-3}, 1.5 \times 10^{-2}]$ with 10 stochastic samples and a pseudo-label NLL-change threshold of 3×10^{-3}.

All foundation-model question-answering experiments were conducted on an NVIDIA A100 GPU with 40 GB of memory. For image classification, model training was performed on an NVIDIA A100 GPU with 40 GB of memory, while post-hoc sampling and evaluation were conducted on an NVIDIA L40S GPU with 48 GB of memory. Regression experiments were run on an Apple M5 Max 36 GB of unified memory, and toy examples were run on an Apple M2 system with 8 GB of unified memory.

[2] *LENS* uses $\eta = 0.8$, $\sigma_d = 0.15$, and $\tau_g = 10^{-12}$ across all tasks. We use $\alpha_{\min} = 10^{-4}$ for UCI and CIFAR-10 and $\alpha_{\min} = 10^{-6}$ for QA. We use $s_{\max} = 5 \times 10^{-2}$ for CIFAR-10 and QA and $s_{\max} = 5 \times 10^{-3}$ for UCI.