Perturb and Correct: Post-Hoc Ensembles using Affine Redundancy

Eleanor Quint
Department of Computing
University of Nebraska–Lincoln
Lincoln, NE 68508, USA
equint@cse.unl.edu

Abstract

Models that are nearly indistinguishable on in-distribution data can behave very differently under distribution shift. We introduce Perturb-and-Correct (P&C), a post-hoc method for constructing epistemically diverse predictors from a single pretrained network. P&C applies random hidden-layer perturbations with a least-squares correction in the following affine layer, producing predictors that agree on calibration data while remaining free to disagree away from it. We analyze this mechanism through the post-correction residual that remains in the activation of the layer corrected using calibration data, yielding a leverage-based preservation bound and a covariance interpretation of ensemble disagreement. Empirically, P&C achieves a strong ID/OOD tradeoff across MuJoCo dynamics prediction and CIFAR-10 OOD detection, and scales to a pretrained ViT-B/16 on ImageNet-1K. In matched comparisons with Deep Ensembles, P&C substantially reduces construction and storage cost. Our findings highlight the potential in further exploiting overparameterization as a strength of deep learning models.

1 Introduction

Modern neural networks are massively overparameterized. Many distinct parameter settings can realize models that are nearly indistinguishable on the data on which they are trained [D'Amour et al., 2022]. This flexibility forms a foundation of the modern deep learning regime, supporting effective first-order optimization [Dauphin et al., 2014, Choromanska et al., 2015] and generalization phenomena distinct from less parameterized models, such as double descent [Belkin et al., 2019]. At the same time, it contributes to a broader problem of *underspecification*, where a training pipeline can return predictors with essentially identical in-distribution (ID) performance even though those predictors may behave very differently under distribution shift [D'Amour et al., 2022]. In such settings, the observed data do not fully determine model behavior away from their support, making epistemic uncertainty and out-of-distribution detection central to reliable deployment.

In this work, we introduce *Perturb-and-Correct* (P&C), a post-hoc ensemble construction for exposing this unresolved model variation (Figure 1). P&C samples random hidden-layer parameter perturbations and then fits a compensating correction in the following affine layer to reproduce the base model's next-layer pre-activations on ID calibration data. The correction suppresses perturbation effects that can be absorbed by the calibration geometry while leaving members free to differ where that repair no longer transfers. In this way, P&C turns a single completed model into a family of nonlinear predictors that agree where calibration data constrain behavior and can disagree where they do not.

Our theory formalizes this intuition by analyzing what remains after a hidden-layer perturbation is corrected. We derive an exact post-correction residual in terms of the nonlinear representation changes induced by the applied perturbation. The residual separates the test-point change from the

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

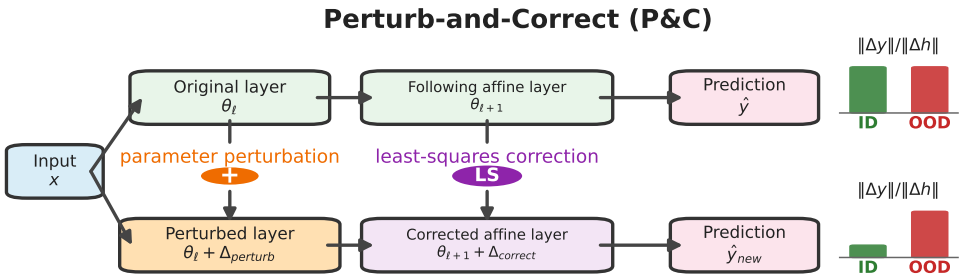


Figure 1: **Perturb-and-Correct.** A hidden-layer perturbation changes the representation h to h_v. The following affine layer is refitted on ID calibration examples so that the corrected predictor remains close to the base model there. At a shifted input, the representation change may not be reconstructed from the changes observed on calibration data, leaving a post-correction residual that can produce ensemble disagreement.

component reconstructed from calibration changes, yielding a leverage-based preservation bound and a *transfer defect*: the portion of the perturbation effect that the calibration geometry fails to reconstruct. Near well-covered calibration representations, the repair can continue to suppress the perturbation; away from that geometry, the transfer defect can remain and produce disagreement. Random P&C therefore defines a calibration-dependent random function whose covariance records shared, unresolved correction failures. The corrected-sensitivity expression is recovered by differentiating this exact object at the unperturbed model, while the exact residual describes the perturbation scales used by the ensemble.

A common strategy in post-hoc uncertainty estimation is to place substantial structure on the source of model variation, for example by sampling an approximate posterior or restricting variation to an SGD-derived subspace. P&C instead places its main structural constraint on the correction. The perturbations are random and low-dimensional; the least-squares repair enforces agreement on calibration data and lets disagreement emerge where that repair no longer transfers. Figure 2 previews this mechanism empirically: corrected P&C achieves a favorable ID/OOD frontier while maintaining large hidden perturbations whose output effect is suppressed on ID data and re-emerges under shift.

We evaluate P&C on probabilistic MuJoCo dynamics prediction and CIFAR-10 OOD detection, and scale the same construction to a pretrained 86.6M-parameter ViT-B/16 on ImageNet-1K. Across these settings, P&C preserves ID behavior while exposing useful disagreement under shift, although arbitrary shift need not increase uncertainty. An additional DistilBERT experiment tests the same construction under semantic shift. In our matched Deep Ensemble comparisons, P&C substantially reduces the cost of constructing and storing the ensemble. Member-dependent inference remains $O(M)$, although computation before the earliest perturbed layer can be shared.

This work makes three contributions. First, we show that useful post-hoc epistemic ensembles can be constructed from a completed checkpoint by pairing random low-dimensional hidden-layer perturbations with local affine correction on calibration data. Second, we derive an exact account of the post-correction residual, including its transfer geometry, a leverage-based preservation bound, a covariance interpretation of ensemble disagreement, and corrected sensitivity as the infinitesimal limit. Third, we provide empirical evidence supporting this mechanism across dynamics, vision, and text, including matched frozen-checkpoint comparisons, sensitivity analyses, and explicit cost accounting. Code is available at `https://github.com/EtherealEq/perturb-and-correct`.

2 Related Work

Ensembles and approximate posterior sampling. A standard approach to epistemic uncertainty estimation is to aggregate predictions across multiple plausible predictors. Deep Ensembles remain a strong empirical baseline, with diversity arising from independent training runs [Lakshminarayanan et al., 2017], while MC Dropout obtains ensemble-like predictions by sampling stochastic subnetworks at test time [Gal and Ghahramani, 2016]. Other post-training approaches construct approximate distributions around a trained solution. The Laplace approximation uses local curvature [Ritter et al.,

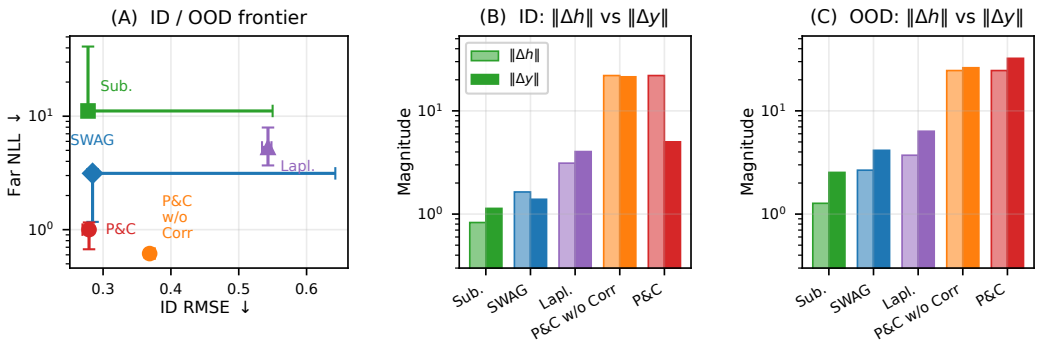


Figure 2: **P&C improves the ID/OOD tradeoff on Ant-v5 by preserving large hidden perturbations while reducing their ID-visible effect. (A)** ID RMSE versus Far-OOD NLL for Subspace, SWAG, Laplace, uncorrected random perturbations, and corrected P&C. **(B–C)** Mean hidden change $\|\Delta h\|$ and output change $\|\Delta y\|$ on ID and Far-OOD data. Correction leaves the hidden perturbation large while suppressing the component of its output effect that transfers through the ID calibration geometry.

2018], SWAG fits a Gaussian approximation to an SGD trajectory [Maddox et al., 2019], and Subnetwork Linearized Laplace places a full-covariance Gaussian over selected internal weights in a linearized model [Daxberger et al., 2021]. P&C also represents uncertainty through an ensemble of predictors, but differs by constructing members from one trained checkpoint using hidden perturbations followed by calibration-preserving correction.

Low-loss geometry and structured parameter-space uncertainty. A related line of work studies structured high-performing regions in neural network parameter space. SWAG exploits the geometry traced by SGD iterates, while Subspace Inference performs posterior inference in a low-dimensional subspace often derived from optimization trajectories [Maddox et al., 2019, Izmailov et al., 2020]. These methods show that useful model variation can often be found in structured low-dimensional regions of parameter space. P&C shares the goal of obtaining uncertainty from low-dimensional parameter variation, but places the main inductive bias on what happens after the perturbation is applied: the perturbation is sampled randomly and the following affine layer is repaired on calibration data. The correcting step reduces the uncertainty to what can't be predicted from calibration data.

Single-model and frozen-checkpoint uncertainty. Several methods seek useful uncertainty estimates without training a full independent ensemble. SNGP combines spectral normalization with a Gaussian-process output layer [Liu et al., 2020a], while Epistemic Neural Networks represent joint predictions through explicit epistemic indexing [Osband et al., 2023]; both require changes to the model or training procedure. BatchEnsemble and Masksembles similarly introduce parameter sharing or structured masks during training [Wen et al., 2020, Durasov et al., 2021]. A different class of methods operates directly on a completed predictor, including MSP, Energy, Mahalanobis distance, and ReAct [Hendrycks and Gimpel, 2016, Liu et al., 2020b, Lee et al., 2018, Sun et al., 2021]. SCOD is especially relevant here: it sketches the training-data Fisher and measures test-point sensitivity along weakly constrained weight directions [Sharma et al., 2021]. P&C is also applied after training, but returns nonlinear predictors and an ensemble predictive distribution rather than a single uncertainty score.

Adversarial sensitivity and constrained variation. The motivation underlying P&C is also related to the adversarial example literature. Adversarial attacks expose the fact that a perturbation can be small under one set of constraints while producing a large predictive effect under another [Goodfellow et al., 2015, Fawzi et al., 2018], and robust optimization makes this asymmetry explicit through constrained worst-case variation [Madry et al., 2018]. P&C uses a related constrained-variation perspective in parameter space. Random perturbations are allowed, while affine repair keeps their ID-visible effects small on calibration data.

3 Perturb-and-Correct

P&C constructs an ensemble from a single trained network by perturbing one or more hidden-layer parameters and refitting a compensating correction in the following affine layer on calibration data. We treat the resulting predictors as a uniformly weighted mixture and combine predictions as in Lakshminarayanan et al. [2017]. Throughout, let

$$f_\theta = g_L \circ \phi \circ g_{L-1} \circ \cdots \circ \phi \circ g_1,$$

where each $g_\ell(h) = W_\ell h + b_\ell$ is affine and ϕ is any fixed nonlinearity.[1] Let $\mathcal{D}_{\text{cal}} = \{x_i\}_{i=1}^N$ denote an ID calibration set, which is drawn from the data used to fit the base network.

Single-layer P&C. Fix a target hidden layer ℓ, ensemble size M, perturbation scale $\sigma > 0$, and perturbation rank K. Draw a random orthonormal basis $U_\ell \in \mathbb{R}^{d_\ell \times K}$ once and share it across members, where d_ℓ is the number of perturbed parameters. For each member $m = 1, \ldots, M$, sample $z^{(m)} \sim \mathcal{N}(0, I_K)$ and form

$$\Delta W_\ell^{(m)} = \sigma \ \text{reshape} \cdot (U_\ell z^{(m)}), \qquad \widetilde{W}_\ell^{(m)} = W_\ell + \Delta W_\ell^{(m)}.$$

Applying $\widetilde{W}_\ell^{(m)}$ changes the post-activation at layer ℓ, which would otherwise propagate through the rest of the network. P&C compensates locally by replacing the next affine layer $(W_{\ell+1}, b_{\ell+1})$ with a corrected pair $(\widetilde{W}_{\ell+1}^{(m)}, \widetilde{b}_{\ell+1}^{(m)})$ fit on calibration activations. Let $h_\ell(x)$ and $\widetilde{h}_\ell^{(m)}(x)$ denote the original and perturbed post-activations, and let $S_m \subseteq \mathcal{D}_{\text{cal}}$ denote the correction subset for member m. The correction solves

$$\begin{aligned}\left(\widetilde{W}_{\ell+1}^{(m)}, \widetilde{b}_{\ell+1}^{(m)}\right) = \arg\min_{W,b} \sum_{x \in S_m} \left\| W\widetilde{h}_\ell^{(m)}(x) + b - \left(W_{\ell+1}h_\ell(x) + b_{\ell+1}\right) \right\|_2^2 \\ + \lambda \|W - W_{\ell+1}\|_F^2 + \lambda \|b - b_{\ell+1}\|_2^2.\end{aligned} \tag{1}$$

The regression target is the base model's next-layer pre-activation rather than a ground-truth label. The ridge penalty keeps the repair close to the pretrained affine map while allowing the calibration data to determine how much compensation is required. Each member shares every remaining parameter with the base model, so the only learned change after perturbation is the local affine repair.

Multi-layer P&C. When perturbing a sequence of layers $\ell_1 < \cdots < \ell_r$, P&C applies the perturb-and-correct step sequentially. After perturbing layer ℓ_j, it fits the correction in layer $\ell_j + 1$ using activations produced by all preceding perturb-and-correct operations before proceeding to ℓ_{j+1}. This yields a chain of local repairs rather than a single end-of-network correction. We perturb hidden layers only and leave output heads unchanged.

Bootstrapped correction. The correction subsets $\{S_m\}_{m=1}^M$ need not be shared across members. P&C can introduce a second source of ensemble diversity by drawing each S_m independently with replacement from $\mathcal{D}_{\text{cal}}$, so that different members solve Eq. (1) on different empirical correction geometries. Changing these subsets can decorrelate correction failures and increase the effective rank of the ensemble covariance, as discussed in Appendix A.6. At the same time, reducing the correction set can make the least-squares problem unstable near its interpolation threshold. We therefore treat bootstrap fraction as controlling both diversity and correction stability.

4 Theory

We analyze P&C as a randomized post-hoc ensemble mechanism. The central object is the *post-correction residual*: the change at a test point that remains after the perturbed representation has been repaired by refitting the following affine layer on calibration data. We show (i) that this residual admits an exact decomposition under the applied perturbation; (ii) that its magnitude is controlled by a leverage term measuring how well the perturbed test representation is covered by the correction design; and (iii) that the local corrected-sensitivity expression and a covariance kernel for ensemble disagreement follow from the same residual. Together, these results give a geometric account of when correction continues to suppress a perturbation and when its effect can re-emerge away from the calibration geometry.

[1] We present P&C for a fully connected network for notational simplicity; Section 5 applies it to convolutional and transformer blocks, and Appendix B gives the corresponding architecture-specific forms.

4.1 Setup and exact correction

Fix a trained network and a hidden layer whose post-activation representation is $y(x) \in \mathbb{R}^d$. Write the following affine layer in augmented form as

$$z(x) = \Theta \overline{y}(x), \qquad \overline{y}(x) := \begin{bmatrix} 1 \\ y(x) \end{bmatrix} \in \mathbb{R}^{d+1}, \qquad \Theta := [b, W]. \tag{2}$$

A hidden-parameter perturbation v replaces $\overline{y}$ by $\overline{y}_v$. Let $S = \{x_i\}_{i=1}^B$ denote a correction subset, and let $\overline{Y}_S, \overline{Y}_{v,S}$ stack the original and perturbed augmented representations respectively in rows, giving

$$\widehat{\Theta}_S(v) := \arg\min_{\Theta'} \left\| \overline{Y}_{v,S} \Theta'^{\top} - \overline{Y}_S \Theta^{\top} \right\|_F^2 + \lambda \|\Theta' - \Theta\|_F^2. \tag{3}$$

Define

$$G_{v,S} := \overline{Y}_{v,S}^{\top} \overline{Y}_{v,S} + \lambda I, \qquad \Delta Y_{v,S} := \overline{Y}_{v,S} - \overline{Y}_S, \qquad \Delta \overline{y}_v(x) := \overline{y}_v(x) - \overline{y}(x),$$

and the post-correction residual

$$r_S(x; v) := \widehat{\Theta}_S(v) \overline{y}_v(x) - \Theta \overline{y}(x), \qquad \rho_S(x; v) := \|r_S(x; v)\|_2. \tag{4}$$

Lemma 1 (Exact correction and residual identity)**.** *If $G_{v,S}$ is nonsingular,*

$$\widehat{\Theta}_S(v) = \Theta - \Theta \Delta Y_{v,S}^{\top} \overline{Y}_{v,S} G_{v,S}^{-1}. \tag{5}$$

With

$$\alpha_{v,S}(x) := \overline{Y}_{v,S} G_{v,S}^{-1} \overline{y}_v(x) \tag{6}$$

the residual is exactly

$$r_S(x; v) = \Theta \Big[\Delta \overline{y}_v(x) - \Delta Y_{v,S}^{\top} \alpha_{v,S}(x) \Big]. \tag{7}$$

Equation (7) makes the action of the correction explicit. The test-point representation changes by $\Delta \overline{y}_v(x)$, while the least-squares fit reconstructs part of that change from the perturbation effects observed on the correction set. What remains after applying the following affine layer is the post-correction residual. Importantly, this identity uses the actual nonlinear representation changes induced by the perturbation rather than a Taylor expansion in v, so it describes the implemented correction without assuming a fixed activation region or first-order accuracy. Appendix A gives the proof and the pseudoinverse extension for rank-deficient unregularized designs.

4.2 ID preservation and the transfer defect

Write the rows of $\Delta Y_{v,S}$ as $\Delta \overline{y}_v(x_i)^{\top}$. The bracketed quantity in Eq. (7) is the *transfer defect*

$$g_S(x; v) := \Delta \overline{y}_v(x) - \sum_{i=1}^{B} \alpha_{v,S,i}(x) \Delta \overline{y}_v(x_i), \qquad r_S(x; v) = \Theta g_S(x; v). \tag{8}$$

The transfer defect measures how much of the test-point representation change is not reconstructed from the changes observed on the correction set under the same least-squares geometry. When calibration examples predict the perturbation effect at x well, $g_S(x; v)$ is small and the repair continues to suppress the perturbation. When this relationship fails to transfer, the residual can remain large. Thus P&C does not require every OOD input to produce high disagreement: an OOD point can still have low disagreement when its perturbation effects remain predictable from calibration data.

Define the perturbed-design ridge leverage

$$h_{v,S}^{\lambda}(x) := \overline{y}_v(x)^{\top} G_{v,S}^{-1} \overline{y}_v(x), \tag{9}$$

and perturbation magnitudes $\Delta_S(v) := \|\Delta Y_{v,S}\|_F$ and $\delta_x(v) := \|\Delta \overline{y}_v(x)\|_2$.

Theorem 2 (Post-correction residual bound)**.** *Under Lemma 1,*

$$\rho_S(x; v) \le \|\Theta\|_{\mathrm{op}} \Big[\delta_x(v) + \Delta_S(v) \sqrt{h_{v,S}^{\lambda}(x)} \Big]. \tag{10}$$

Moreover,

$$\|\alpha_{v,S}(x)\|_2^2 = h_{v,S}^{\lambda}(x) - \lambda \|G_{v,S}^{-1} \overline{y}_v(x)\|_2^2 \le h_{v,S}^{\lambda}(x). \tag{11}$$

The appearance of leverage gives the bound its geometric interpretation. When the perturbed representation $\overline{y}_v(x)$ is well covered by the correction design, $h^{\lambda}_{v,S}(x)$ is controlled and so are the reconstruction weights $\alpha_{v,S}(x)$. If the perturbation also changes nearby calibration representations in a way that predicts the change at x, the post-correction residual remains small. As leverage grows or this transfer relationship breaks, the bound becomes correspondingly less restrictive. Unlike the local sensitivity analysis below, the bound is pointwise in the applied perturbation and contains no Taylor remainder.

4.3 Corrected sensitivity as the local limit

The exact transfer defect also recovers the corrected-sensitivity decomposition studied in the local analysis. Suppose the representation maps are differentiable at $v = 0$. Let

$$J_x := \partial_v \overline{y}_v(x)|_{v=0}, \qquad w^{\lambda}_S(x) := \overline{Y}_S(\overline{Y}_S^{\top}\overline{Y}_S + \lambda I)^{-1}\overline{y}(x),$$

and define the calibration-predicted Jacobian

$$T^{\lambda}_S(x) := \sum_{i=1}^{B} w^{\lambda}_{S,i}(x) J_{x_i}.$$

Corollary 3 (Corrected sensitivity)**.** *The derivative $A_S(x) := \partial_v r_S(x; v)|_{v=0}$ satisfies*

$$A_S(x) = \Theta\big(J_x - T^{\lambda}_S(x)\big). \tag{12}$$

For $v \sim \mathcal{N}(0, \sigma^2 I)$, $\mathbb{E}\|A_S(x)v\|_2^2 = \sigma^2 \|A_S(x)\|_F^2$.

This expression gives the same local geometric picture: corrected sensitivity is small when calibration Jacobians predict the test-point Jacobian and grows where that prediction fails. Random directions therefore probe the part of local sensitivity left unresolved by the calibration geometry without requiring knowledge of the eventual test-time shift. At the perturbation scales used in our experiments, however, the derivative at zero is not quantitatively accurate for the full member change. We therefore use the exact residual for the applied perturbations and treat Eq. (12) as its infinitesimal limit.

4.4 Random perturbations and ensemble disagreement

Let V denote member randomness, including perturbation coefficients and, when used, the bootstrap correction subset. For each input let $\mu_S(x) := \mathbb{E}_V r_V(x)$ and $\widetilde{r}_V(x) := r_V(x) - \mu_S(x)$, and define

$$K_S(x, x') := \mathbb{E}_V[\widetilde{r}_V(x)\widetilde{r}_V(x')^{\top}]. \tag{13}$$

Theorem 4 (Random-function covariance kernel)**.** *If $\mathbb{E}_V\|r_V(x)\|_2^2 < \infty$, then K_S is a positive-semidefinite matrix-valued kernel. Its trace $k_S(x, x') := \operatorname{tr} K_S(x, x')$ is a positive-semidefinite scalar kernel, and the usual sample covariance of i.i.d. P&C members is an unbiased estimator of K_S.*

The kernel describes how unresolved correction failures vary across the input space. Its diagonal $k_S(x, x)$ is the residual variance exposed by P&C at x, while off-diagonal values identify points whose output-visible transfer defects co-vary across perturbations. In particular,

$$\mathbb{E}_V\|r_V(x)\|_2^2 = \|\mu_S(x)\|_2^2 + \operatorname{tr} K_S(x, x), \qquad \mathbb{E}_{V,V'}\|r_V(x) - r_{V'}(x)\|_2^2 = 2\operatorname{tr} K_S(x, x), \tag{14}$$

for independent members V, V'. Pairwise ensemble disagreement is therefore exactly twice the diagonal mass of this covariance kernel. Appendix A.4 gives the proof, and Appendix A.6 discusses how changing correction subsets can affect covariance rank.

5 Experiments

5.1 Experimental setup

We evaluate Perturb-and-Correct (P&C) in three main settings: probabilistic dynamics prediction under behavioral distribution shift in MuJoCo [Todorov et al., 2012, Towers et al., 2024], out-of-distribution (OOD) detection on CIFAR-10 [Krizhevsky and Hinton, 2009] under the OpenOOD

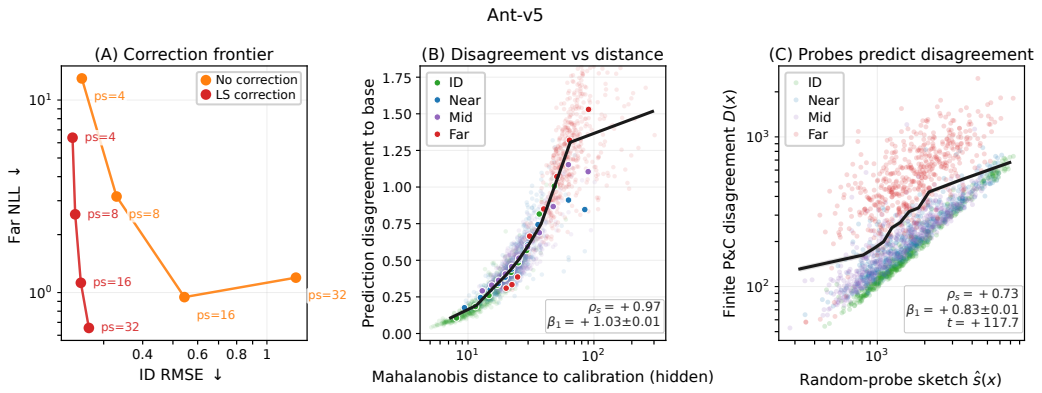


Figure 3: **Mechanism-level diagnostics for P&C on Ant-v5. (A)** Least-squares correction improves the ID/OOD frontier: uncorrected random perturbations degrade ID behavior, while corrected P&C achieves lower Far-OOD NLL at comparable ID error. **(B)** Prediction disagreement increases with Mahalanobis distance from the ID calibration activations. **(C)** The random-probe sketch from the infinitesimal corrected-sensitivity analysis remains correlated with the disagreement of the actual P&C ensemble (Spearman $\rho_s = 0.73$), although the local approximation is not quantitatively accurate at the experimental perturbation scale. Together, these diagnostics support the geometric picture in Section 4: correction suppresses perturbation effects that are predictable from calibration data, while disagreement emerges where that relationship fails to transfer.

v1.5 protocol [Zhang et al., 2023], and ImageNet-1K OOD detection [Deng et al., 2009] with a pretrained ViT-B/16 [Dosovitskiy et al., 2020]. Across all settings, P&C is constructed post hoc from a completed model, and all hyperparameters and post-hoc calibration choices are selected using in-distribution (ID) validation data only. We additionally report a DistilBERT Banking77 experiment in Appendix F as a modality-transfer check.

MuJoCo uses 4,096 correction examples from the expert-transition training pool, CIFAR-10 uses 1,024 images from the training-fit split, and ImageNet uses 32,768 training images for correction with separate ID pools for selection and temperature fitting. Test and OOD examples never enter construction or selection. MuJoCo members are combined by moment matching their diagonal Gaussian predictions, with mean total predictive variance used for OOD ranking. For the classification experiments, we fit a scalar temperature on ID data, average member probabilities, and use predictive entropy as the P&C OOD score. Full architecture, calibration, selection, and evaluation protocols are given in the appendices.

5.2 Validating the Perturb-and-Correct mechanism

Before turning to aggregate benchmarks, we validate the mechanism described in Section 4. The exact residual identity in Eq. (7) holds numerically to approximately 10^{-13} relative error in float64 on Ant, HalfCheetah, and Hopper across ID and shifted inputs. Propagating the corrected-layer residual through the frozen downstream Jacobian yields cosine alignment 0.97–1.00 with the final output change. By contrast, the derivative-at-zero approximation is accurate only at perturbation norms roughly 30–$300\times$ smaller than the operating range. We therefore use the exact residual to describe the applied perturbations and retain corrected sensitivity as its local limit. Appendix C.3 gives the validation protocol.

Figure 3(A) compares random perturbations with and without least-squares correction on Ant-v5. Without correction, perturbations damage ID behavior; adding the repair shifts the frontier substantially, preserving ID performance without eliminating disagreement under Far-OOD shift. Figure 3(B) shows the corresponding geometric pattern: P&C disagreement increases with hidden Mahalanobis distance from the calibration activations. This relationship is not produced only by pooling ID and OOD regimes. On 40,000 points per environment, HalfCheetah and Hopper have pooled Spearman correlations 0.904 and 0.749, with within-tier correlations 0.816–0.860 and 0.691–0.776 respectively.

Table 1: Cross-environment summary across 11 MuJoCo environments and 10 seeds. Entries report mean rank $\pm$ stddev and win counts; lower rank is better. Header arrows indicate the preferred direction of the underlying metric; superscript arrows indicate Wilcoxon-Holm comparisons against P&C at $\alpha = 0.05$: $\uparrow$ means P&C significantly outperforms the baseline, and $\downarrow$ means the reverse.

Method	ID RMSE ↓		Far NLL ↓		Far AUROC ↑		Far Spearman ↑	
	mean rank	wins	mean rank	wins	mean rank	wins	mean rank	wins
Deep Ensemble	3.27 ± 1.56	2 / 11	3.00 ± 1.18	1 / 11	$2.64 \pm 1.36^{\uparrow}$	1 / 11	2.27 ± 1.35	4 / 11
MC Dropout	3.82 ± 1.40	0 / 11	3.45 ± 1.13	0 / 11	$3.18 \pm 1.72^{\uparrow}$	1 / 11	$4.18 \pm 1.40^{\uparrow}$	0 / 11
SWAG	4.55 ± 1.44	0 / 11	4.00 ± 1.55	1 / 11	$4.64 \pm 1.21^{\uparrow}$	0 / 11	$4.91 \pm 0.70^{\uparrow}$	0 / 11
Laplace	3.18 ± 1.47	1 / 11	3.09 ± 1.14	1 / 11	$4.18 \pm 0.87^{\uparrow}$	0 / 11	$2.91 \pm 0.83^{\uparrow}$	0 / 11
Subspace	$\mathbf{3.00 \pm 1.73}$	3 / 11	$5.91 \pm 0.30^{\uparrow}$	0 / 11	$5.09 \pm 0.83^{\uparrow}$	0 / 11	$5.09 \pm 1.38^{\uparrow}$	0 / 11
P&C	3.18 ± 2.40	5 / 11	$\mathbf{1.55 \pm 1.21}$	8 / 11	$\mathbf{1.27 \pm 0.65}$	9 / 11	$\mathbf{1.64 \pm 1.03}$	7 / 11

Finally, Figure 3(C) connects the local corrected-sensitivity analysis to the actual ensemble. Although the derivative at zero is not quantitatively accurate at the applied perturbation scale, its random-probe sketch remains correlated with P&C disagreement. Random-vs-Low diagnostics in Appendix C.4 further support random perturbation directions across Ant, HalfCheetah, and Hopper. Together, these results support the view that correction, rather than carefully chosen perturbation directions, is the main mechanism shaping useful disagreement.

5.3 MuJoCo: uncertainty under behavioral shift

We evaluate multi-block P&C on dynamics prediction across 11 MuJoCo environments. Models are evaluated on held-out expert transitions and progressively shifted policy data; Far transitions use uniform-random actions, with Near and Mid tiers available where provided by the benchmark. No shifted data are used for construction, selection, or calibration. Full per-environment results and hyperparameter details are in Appendix C.

Table 1 summarizes the comparison by cross-environment rank. P&C is competitive on ID RMSE, with mean rank 3.18 ± 2.40 and five wins, and obtains the best mean rank on Far-OOD NLL (1.55 ± 1.21), AUROC (1.27 ± 0.65), and uncertainty–error Spearman correlation (1.64 ± 1.03), with $8/11$, $9/11$, and $7/11$ wins respectively. The environment-level results are not uniformly favorable: ID error is mixed, HalfCheetah has notable variance, and Humanoid remains difficult. In a separate 27-seed graded-shift study, Far-AUROC reaches $0.998/0.987/0.980$ on Ant, HalfCheetah, and Hopper but falls to 0.343 on Humanoid. Because this study uses a different run and aggregation protocol from the 10-seed validation-selected comparison, Appendix C reports the two separately.

We also compare P&C with SCOD from matched frozen checkpoints. P&C has higher Far-AUROC in all 11 environments, with mean 0.920 versus 0.512 over the three matched seeds. A 27-seed layer-scope study gives Near/Mid/Far AUROC $0.764/0.815/0.904$ for first-block P&C and $0.799/0.843/0.922$ for the multi-block construction. Multi-block correction is inexpensive for these small MLPs; the convolutional experiments instead select a single target block by ID-validation NLL to avoid compounding larger patchwise solves. Across the MuJoCo sensitivity sweeps, performance is broadly stable to rank and ridge, while perturbation scale exhibits an ID-stability boundary and correction-set size shows interpolation effects near the affine design dimension.

P&C also retains the construction-cost advantage that motivates a post-hoc ensemble. For $M = 50$, it is 46.8–$61.7\times$ cheaper to construct than a matched Deep Ensemble on the three timed MuJoCo environments, adding only 1.5–1.8 seconds beyond base training. This advantage concerns construction rather than inference: both methods still require M member evaluations at test time.

5.4 CIFAR-10: OOD detection

We next evaluate single-block P&C on CIFAR-10 using a pre-activation ResNet-18 [He et al., 2016] under the OpenOOD v1.5 protocol. Near OOD comprises CIFAR-100 and TinyImageNet, while Far OOD comprises MNIST, SVHN, Textures, and Places365; this protocol does not define a separate Mid tier. We compare against Mahalanobis, ReAct+Energy, LLLA, Epinet, SWAG, a Deep Ensemble, and matched SCOD. Target block, perturbation scale, and bootstrap fraction are selected by ID-validation NLL only.

Table 2: CIFAR-10 OOD detection under OpenOOD v1.5 over three independently trained checkpoints (mean $\pm$ population standard deviation). Near OOD comprises CIFAR-100 and TinyImageNet; Far OOD comprises MNIST, SVHN, Textures, and Places365. SCOD uses the same checkpoints and a matched 1,024-example ID calibration pool. Bold and italic mark the best and second-best values.

Method	Acc% ↑	Near AUROC ↑	Near FPR95 ↓	Far AUROC ↑	Far FPR95 ↓
Mahalanobis	95.74 ± 0.14	87.98 ± 0.11	66.30 ± 0.88	93.25 ± 0.26	38.30 ± 5.44
ReAct+Energy	*95.77 ± 0.16*	88.63 ± 0.15	54.39 ± 1.28	92.50 ± 1.20	29.78 ± 5.26
LLLA	*95.77 ± 0.20*	88.97 ± 0.69	54.12 ± 5.70	93.04 ± 0.95	28.44 ± 6.72
Epinet	95.76 ± 0.18	88.07 ± 0.04	63.37 ± 1.40	92.16 ± 1.02	35.10 ± 5.57
SWAG	95.37 ± 0.05	90.03 ± 0.15	44.71 ± 0.79	94.19 ± 1.03	22.09 ± 4.91
SCOD	95.74 ± 0.14	89.69 ± 0.09	*39.40 ± 0.96*	92.56 ± 0.37	21.41 ± 0.87
P&C	95.59 ± 0.20	$\mathbf{91.55 \pm 0.13}$	$\mathbf{33.08 \pm 0.63}$	$\mathbf{95.09 \pm 0.50}$	$\mathbf{18.15 \pm 1.07}$
Deep Ensemble	$\mathbf{96.56 \pm 0.04}$	*91.10 ± 0.03*	40.39 ± 0.38	*94.63 ± 0.14*	*19.49 ± 0.26*

Table 2 shows that P&C obtains the highest Near- and Far-OOD AUROC and the lowest corresponding FPR95 among the reported methods. Against matched SCOD, Near/Far AUROC improves from $89.69/92.56$ to $91.55/95.09$, while FPR95 falls from $39.40/21.41$ to $33.08/18.15$. The five-model Deep Ensemble retains the highest ID accuracy. As in MuJoCo, the main effect of P&C is not to improve the base model's ID prediction, but to preserve that prediction while introducing model variation that becomes useful under shift.

We test whether these results depend on a narrow hyperparameter choice using a complete Cartesian product of six target blocks, three perturbation scales, and three bootstrap fractions over all three checkpoints, for 162 runs. Global ID-validation search and forward or reverse coordinate-wise selection recover the same optimum. Separate sweeps show nearly flat rank performance for $K \geq 5$, ridge stability for $\lambda \geq 10^{-3}$ but instability at zero, and a clear perturbation-scale ID-stability boundary (Appendix D). Representative early, middle, and late blocks remain viable.

The base ResNet-18 takes $8{,}010$ seconds to train. P&C adds at most about 100 seconds of post-training construction, while a matched 50-member Deep Ensemble would require approximately $400{,}500$ seconds by repeated base training. At inference, the two methods have essentially equal matched-M latency (7.53 versus 7.26 ms/example). The computational advantage in this comparison is therefore the cost of constructing and storing the ensemble, not the online cost of evaluating it; Appendix D.6 gives the full timing and memory accounting.

5.5 ImageNet-1K: large-scale transformer OOD detection

We next apply P&C to the final feed-forward block of a pretrained 86.6M-parameter ViT-B/16 on ImageNet-1K. P&C perturbs the first affine map of the final transformer FFN and refits the second affine map on final-block CLS representations. Because there is no later attention operation after this FFN, the correction can be fit on CLS rows alone; Appendix B.7 gives the architectural details.

The primary configuration uses $K = 20$ and $M = 20$. Perturbation magnitude and ridge are selected using ID data only under a fixed 0.50 percentage-point top-1 preservation budget, selecting relative perturbation magnitude $r = 2$ and $\lambda = 1000$. The official 50,000-image ImageNet validation set is left untouched until final evaluation. The base checkpoint obtains 81.068% top-1 accuracy, NLL 0.8482, and ECE 0.0913, while P&C obtains 80.830%, 0.8045, and 0.0542, respectively.

Table 3 shows that correction remains important at this scale. The same $r = 2$ perturbations without correction obtain Near/Far AUROC $72.53/84.40$, while corrected P&C obtains $76.49/87.89$, improvements of 3.96 and 3.49 points. P&C also exceeds MSP on both aggregates and LLLA-Kron on Near OOD; against SCOD-linear it is stronger on Near and weaker on Far. A Mahalanobis detector on the final normalized CLS representation remains stronger for pure OOD ranking. Appendix E gives the preservation frontier, matched ablations, efficiency accounting, and additional mechanism diagnostics.

Limitations. These experiments do not establish universal effectiveness. P&C depends on the calibration set covering the intended ID behavior, on the correction design remaining sufficiently well conditioned, and on perturbations staying within an ID-stable range. It can also be limited by the choice of target layer and the amount of useful downstream affine redundancy. More generally,

Table 3: ImageNet-1K OOD detection with a pretrained ViT-B/16. Near OOD comprises SSB-hard and NINCO; Far OOD comprises iNaturalist, Textures, and OpenImage-O. P&C uses the primary 0.50 percentage-point ID-preservation budget, selecting $r = 2$ and $\lambda = 1000$ using ID data only. Uncorrected perturbation uses the identical perturbations without the affine repair. P&C and Uncorrected are averaged over five post-hoc construction seeds.

Method	Near AUROC ↑	Far AUROC ↑
MSP	73.52	86.04
Mahalanobis	**78.82**	**92.55**
LLLA-Kron	72.95	87.42
SCOD-linear	75.15	*88.48*
Uncorrected perturb.	72.53	84.40
P&C	*76.49*	87.89

arbitrary distribution shift need not increase disagreement, and dedicated representation-space scores can be stronger when the task is pure OOD ranking, as Mahalanobis is on ImageNet. Member-dependent inference remains $O(M)$, although late-layer constructions can reuse the shared network prefix. Broader evaluation across pretrained architectures, language models, and settings that require temporally coherent uncertainty remains important future work.

6 Conclusion

We introduced Perturb-and-Correct (P&C), a post-hoc method for constructing epistemically diverse ensembles from a single pretrained network. By pairing random hidden-layer perturbations with local affine correction, P&C preserves in-distribution behavior while allowing disagreement to emerge where the same behavior fails to transfer. Our analysis makes this mechanism explicit through the exact post-correction residual: calibration changes control for part of the perturbation effect, while the remaining transfer defect controls the output-visible change and the resulting ensemble covariance. Across MuJoCo dynamics prediction, CIFAR-10 OOD detection, and ImageNet-1K with a pretrained ViT-B/16, P&C preserves ID behavior while exposing useful variation under shift; an additional DistilBERT experiment extends the same construction to text. Its matched advantage over Deep Ensembles lies primarily in inexpensive post-hoc construction and compact storage, while member-dependent inference remains $O(M)$ and arbitrary distribution shift need not produce useful disagreement. Within these limits, P&C provides a simple way to expose predictive degrees of freedom left unresolved after preserving a completed model on calibration data.

Acknowledgments and Disclosure of Funding

I would like to acknowledge the support of my partner. This work was not funded.

Competing interests: Within the relevant 36-month period, I was employed by Phaidra.

References

Mikhail Belkin, Daniel Hsu, Siyuan Ma, and Soumik Mandal. Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences*, 116(32):15849–15854, 2019.

Anna Choromanska, Mikael Henaff, Michael Mathieu, Gérard Ben Arous, and Yann LeCun. The loss surfaces of multilayer networks. In *Artificial intelligence and statistics*, pages 192–204. PMLR, 2015.

Alexander D'Amour, Katherine Heller, Dan Moldovan, Ben Adlam, Babak Alipanahi, Alex Beutel, Christina Chen, Jonathan Deaton, Jacob Eisenstein, Matthew D Hoffman, et al. Underspecification presents challenges for credibility in modern machine learning. *Journal of Machine Learning Research*, 23(226):1–61, 2022.

Yann N Dauphin, Razvan Pascanu, Caglar Gulcehre, Kyunghyun Cho, Surya Ganguli, and Yoshua Bengio. Identifying and attacking the saddle point problem in high-dimensional non-convex optimization. *Advances in neural information processing systems*, 27, 2014.

Erik Daxberger, Eric Nalisnick, James U Allingham, Javier Antorán, and José Miguel Hernández-Lobato. Bayesian deep learning via subnetwork inference. In *International Conference on Machine Learning*, pages 2510–2521. PMLR, 2021.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Nikita Durasov, Timur Bagautdinov, Pierre Baque, and Pascal Fua. Masksembles for uncertainty estimation. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13534–13543. IEEE, 2021.

Alhussein Fawzi, Hamza Fawzi, and Omar Fawzi. Adversarial vulnerability for any classifier. *Advances in neural information processing systems*, 31, 2018.

Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059. PMLR, 2016.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*, 2015.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *European conference on computer vision*, pages 630–645. Springer, 2016.

Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *arXiv preprint arXiv:1610.02136*, 2016.

Pavel Izmailov, Wesley J Maddox, Polina Kirichenko, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. Subspace inference for bayesian deep learning. In *Uncertainty in Artificial Intelligence*, pages 1169–1179. PMLR, 2020.

Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. Technical report, Computer Science Department, University of Toronto, Toronto, Ontario, Canada, 2009.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems*, 30, 2017.

Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. *Advances in neural information processing systems*, 31, 2018.

Jeremiah Liu, Zi Lin, Shreyas Padhy, Dustin Tran, Tania Bedrax Weiss, and Balaji Lakshminarayanan. Simple and principled uncertainty estimation with deterministic deep learning via distance awareness. *Advances in neural information processing systems*, 33:7498–7512, 2020a.

Weitang Liu, Xiaoyun Wang, John Owens, and Yixuan Li. Energy-based out-of-distribution detection. *Advances in neural information processing systems*, 33:21464–21475, 2020b.

Wesley J Maddox, Pavel Izmailov, Timur Garipov, Dmitry P Vetrov, and Andrew Gordon Wilson. A simple baseline for bayesian uncertainty in deep learning. *Advances in neural information processing systems*, 32, 2019.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018.

Ian Osband, Zheng Wen, Seyed Mohammad Asghari, Vikranth Dwaracherla, Morteza Ibrahimi, Xiuyuan Lu, and Benjamin Van Roy. Epistemic neural networks. *Advances in Neural Information Processing Systems*, 36:2795–2823, 2023.

Hippolyt Ritter, Aleksandar Botev, and David Barber. A scalable laplace approximation for neural networks. In *International conference on learning representations*, 2018.

Apoorva Sharma, Navid Azizan, and Marco Pavone. Sketching curvature for efficient out-of-distribution detection for deep neural networks. In *Uncertainty in artificial intelligence*, pages 1958–1967. PMLR, 2021.

Yiyou Sun, Chuan Guo, and Yixuan Li. React: Out-of-distribution detection with rectified activations. *Advances in neural information processing systems*, 34:144–157, 2021.

Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ international conference on intelligent robots and systems*, pages 5026–5033. IEEE, 2012.

Mark Towers, Ariel Kwiatkowski, Jordan Terry, John U Balis, Gianluca De Cola, Tristan Deleu, Manuel Goulão, Andreas Kallinteris, Markus Krimmel, Arjun KG, et al. Gymnasium: A standard interface for reinforcement learning environments. *arXiv preprint arXiv:2407.17032*, 2024.

Yeming Wen, Dustin Tran, and Jimmy Ba. Batchensemble: an alternative approach to efficient ensemble and lifelong learning. *arXiv preprint arXiv:2002.06715*, 2020.

Jingyang Zhang, Jingkang Yang, Pengyun Wang, Haoqi Wang, Yueqian Lin, Haoran Zhang, Yiyou Sun, Xuefeng Du, Yixuan Li, Ziwei Liu, et al. OpenOOD v1.5: Enhanced benchmark for out-of-distribution detection. *arXiv preprint arXiv:2306.09301*, 2023.

A Proofs and additional theory for Section 4

Throughout this appendix, abbreviate

$$X := \overline{Y}_S, \quad X_v := \overline{Y}_{v,S}, \quad \Delta X := X_v - X, \quad a := \overline{y}(x), \quad a_v := \overline{y}_v(x), \quad \Delta a := a_v - a,$$

and let $G_v := X_v^\top X_v + \lambda I$.

A.1 Proof of Lemma 1

Proof. Write $B := \Theta'^\top$. The objective in Eq. (3) becomes

$$\|X_v B - X\Theta^\top\|_F^2 + \lambda\|B - \Theta^\top\|_F^2,$$

with normal equations

$$(X_v^\top X_v + \lambda I)B = X_v^\top X\Theta^\top + \lambda\Theta^\top.$$

Using $X = X_v - \Delta X$ gives

$$B = \Theta^\top - G_v^{-1}X_v^\top \Delta X\, \Theta^\top,$$

or, after transposing,

$$\widehat{\Theta}_S(v) = \Theta - \Theta\Delta X^\top X_v G_v^{-1},$$

which is Eq. (5). Multiplying by a_v and subtracting Θa gives Eq. (7). □

Unregularized pseudoinverse extension. When $\lambda = 0$ and X_v is rank deficient, the minimum-norm solution is $\widehat{\Theta}_S(v)^\top = X_v^+ X\Theta^\top$. Let $P_v := X_v^+ X_v$ and $\alpha_v^+(x) := (X_v^+)^\top a_v$. Then

$$r_S(x; v) = \Theta\Big[\Delta a - \Delta X^\top \alpha_v^+(x) - (I - P_v)a_v\Big]. \tag{15}$$

Thus a rank-deficient minimum-norm solve adds the explicit row-space projection defect $(I - P_v)a_v$.

A.2 Proof of Theorem 2

Proof. From Eq. (7),

$$\begin{aligned}\rho_S(x; v) &\le \|\Theta\|_{\mathrm{op}}\big(\|\Delta a\|_2 + \|\Delta X^\top \alpha_{v,S}(x)\|_2\big)\\ &\le \|\Theta\|_{\mathrm{op}}\big(\delta_x(v) + \Delta_S(v)\|\alpha_{v,S}(x)\|_2\big).\end{aligned}$$

Since $X_v^\top X_v = G_v - \lambda I$,

$$\begin{aligned}\|\alpha_{v,S}(x)\|_2^2 &= a_v^\top G_v^{-1} X_v^\top X_v G_v^{-1} a_v\\ &= a_v^\top G_v^{-1} a_v - \lambda a_v^\top G_v^{-2} a_v\\ &= h_{v,S}^\lambda(x) - \lambda\|G_v^{-1}a_v\|_2^2 \le h_{v,S}^\lambda(x).\end{aligned}$$

Substitution proves Eqs. (10) and (11). □

A.3 Transfer-defect identity

The ith row of ΔX is $\Delta\overline{y}_v(x_i)^\top$, so

$$\Delta X^\top \alpha_{v,S}(x) = \sum_{i=1}^{B} \alpha_{v,S,i}(x)\Delta\overline{y}_v(x_i).$$

Substitution into Eq. (7) gives Eq. (8) and $r_S(x; v) = \Theta g_S(x; v)$.

A.4 Covariance-kernel properties

For arbitrary inputs $x_1, \dots, x_n$ and vectors $c_1, \dots, c_n \in \mathbb{R}^q$,

$$\sum_{i,j} c_i^\top K_S(x_i, x_j) c_j = \mathbb{E}_V \left(\sum_i c_i^\top \widetilde{r}_V(x_i) \right)^2 \geq 0,$$

so K_S is a positive-semidefinite matrix-valued kernel. Its trace kernel is PSD by the same argument. For i.i.d. members, the usual sample cross-covariance is unbiased by the standard sample-covariance identity.

Writing $\mu_S(x) = \mathbb{E}_V r_V(x)$ gives

$$\mathbb{E}_V \|r_V(x)\|_2^2 = \|\mu_S(x)\|_2^2 + \operatorname{tr} K_S(x, x).$$

For independent members V, V' the centered cross term vanishes, yielding

$$\mathbb{E}\|r_V(x) - r_{V'}(x)\|_2^2 = 2 \operatorname{tr} K_S(x, x),$$

as stated in Eq. (14).

A.5 Corrected sensitivity as the infinitesimal limit

At $v = 0$, $\Delta a = \Delta X = 0$ and

$$\alpha_{0,S}(x) = X(X^\top X + \lambda I)^{-1} a = w_S^\lambda(x).$$

Differentiating $g_S(x; v) = \Delta a(v) - \Delta X(v)^\top \alpha_{v,S}(x)$ at zero eliminates the derivative of $\alpha_{v,S}$ because it is multiplied by $\Delta X(0) = 0$. Collecting perturbation coordinates gives

$$A_S(x) = \Theta\big(J_x - T_S^\lambda(x)\big),$$

which is Eq. (12). For $v \sim \mathcal{N}(0, \sigma^2 I)$, $\mathbb{E}\|A_S(x)v\|_2^2 = \sigma^2 \|A_S(x)\|_F^2$.

A.6 Bootstrapped correction and covariance alignment

For correction subset S_j, let $C_j(x) := K_{S_j}(x, x) \succeq 0$ and define

$$r_{\text{eff}}(C) := \frac{\operatorname{tr}(C)^2}{\|C\|_F^2}.$$

Theorem 5 (Mixture rank versus covariance alignment)**.** *Let $C_1, \dots, C_M \in \mathbb{R}^{q \times q}$ be PSD matrices satisfying $\operatorname{tr}(C_j) = \tau$ and $\|C_j\|_F = \nu$. Define $\overline{C} = M^{-1} \sum_j C_j$ and*

$$\overline{\alpha} := \frac{1}{M(M-1)} \sum_{i \neq j} \frac{\langle C_i, C_j \rangle_F}{\|C_i\|_F \|C_j\|_F}. \tag{16}$$

Then

$$r_{\text{eff}}(\overline{C}) = \frac{\tau^2}{\nu^2} \frac{M}{1 + (M-1)\overline{\alpha}}, \tag{17}$$

which decreases monotonically with $\overline{\alpha}$.

Proof. Linearity gives $\operatorname{tr}(\overline{C}) = \tau$, while

$$\|\overline{C}\|_F^2 = \frac{\nu^2}{M}\big(1 + (M-1)\overline{\alpha}\big).$$

Substitution into the effective-rank definition proves Eq. (17). □

The equal-trace and equal-norm assumptions need not hold exactly empirically. The theorem is a structural explanation: changing correction geometry can increase covariance rank when transfer failures are weakly aligned, but bootstrapping does not guarantee such decorrelation and can become unstable near the interpolation threshold.

B Algorithmic details for convolutional and transformer networks

Section 3 presents P&C for a fully connected network. The CIFAR-10 experiments apply the same construction to a pre-activation ResNet-18, while the ImageNet experiment applies it to the feed-forward block of a Vision Transformer. In both cases the perturbation is followed by an affine map that can be refit by the same ridge-regression correction. The exact correction and post-correction residual analysis of Section 4 therefore carries over after reinterpreting the correction design for the architecture at hand.

B.1 Convolutions as patchwise linear maps

Let $X \in \mathbb{R}^{N\times H\times W\times C_{\mathrm{in}}}$ be a batch of feature maps and let a convolution have kernel side length k, stride s, and C_{out} output channels. Write the kernel in OIHW order as

$$W_{\mathrm{OIHW}} \in \mathbb{R}^{C_{\mathrm{out}}\times C_{\mathrm{in}}\times k\times k}.$$

Let

$$P_s(X) \in \mathbb{R}^{(NH'W')\times(C_{\mathrm{in}}k^2)}$$

be the unfolded patch matrix, with one receptive field per output position, and let $\mathrm{mat}(W_{\mathrm{OIHW}}) \in \mathbb{R}^{(C_{\mathrm{in}}k^2)\times C_{\mathrm{out}}}$ flatten the kernel in the matching order. Then

$$\mathrm{flat}(\mathrm{Conv}(X, W)) = P_s(X)\,\mathrm{mat}(W_{\mathrm{OIHW}}). \tag{18}$$

A broadcast bias is included by augmenting $P_s(X)$ with a column of ones, exactly as in the augmented representation $\overline{y}$ of Section 4. Implementations storing kernels in HWIO order apply only a fixed permutation before and after the solve.

B.2 Pre-activation residual blocks as perturb–correct units

We use a CIFAR-adapted pre-activation ResNet-18 with a 3×3 stride-1 stem, no max-pool, and four stages of two residual blocks with widths $64, 128, 256, 512$. A basic block has the form

$$y(x) = \mathrm{Conv}_1(\mathrm{ReLU}(\mathrm{BN}_1(x)); W_1), \tag{19}$$
$$u(x) = \mathrm{ReLU}(\mathrm{BN}_2(y(x))), \tag{20}$$
$$z(x) = \mathrm{Conv}_2(u(x); W_2), \tag{21}$$
$$\mathrm{Block}(x) = z(x) + \mathrm{Shortcut}(x). \tag{22}$$

Batch-normalization layers use the running statistics of the trained base model and remain frozen during construction and inference.

For a selected block, P&C perturbs W_1 and refits W_2 together with a new Conv_2 bias. The shortcut is unchanged, so matching the original Conv_2 output is equivalent to matching the residual branch contribution to the block output. All other network parameters remain frozen.

B.3 Single-block P&C

Fix ensemble size M, perturbation scale σ, and rank K. Draw an orthonormal basis $U \in \mathbb{R}^{d_1\times K}$ in the flattened Conv_1 kernel space, where $d_1 = k_1^2 C_{\mathrm{in},1} C_{\mathrm{out},1}$. For member m,

$$\Delta W_1^{(m)} = \sigma\ \mathrm{reshape}(U z^{(m)}), \qquad z^{(m)} \sim \mathcal{N}(0, I_K),$$

and $\widetilde{W}_1^{(m)} = W_1 + \Delta W_1^{(m)}$.

Let $U(x; W_1)$ denote the tensor entering Conv_2 after Conv_1, BN_2, and ReLU, and let T_i^{orig} be the original Conv_2 output for calibration image x_i. For member m, define the unfolded perturbed design

$$Y_i^{(m)} := P_{s_2}(U(x_i; \widetilde{W}_1^{(m)})), \qquad \overline{Y}_i^{(m)} = [\mathbf{1}, Y_i^{(m)}].$$

Writing Θ for the matrix that stacks the new bias and flattened Conv_2 kernel, and Θ_0 for the original Conv_2 kernel augmented with zero bias, the correction is

$$\widehat{\Theta}^{(m)} = \arg\min_{\Theta} \sum_{i\in S_m} \left\| \overline{Y}_i^{(m)}\Theta - T_i^{\mathrm{orig,flat}} \right\|_F^2 + \lambda\|\Theta - \Theta_0\|_F^2. \tag{23}$$

Thus the CIFAR implementation uses ridge centered on the original affine map.

Stacking rows over the member's correction subset gives $\overline{Y}_{S_m}^{(m)}$ and $T_{S_m}^{\mathrm{orig}}$. With

$$H^{(m)} = \overline{Y}_{S_m}^{(m)\top} \overline{Y}_{S_m}^{(m)},$$

the closed-form correction can be written as

$$\widehat{\Theta}^{(m)} = \left(H^{(m)} + \lambda I\right)^{-1} \left(\overline{Y}_{S_m}^{(m)\top} T_{S_m}^{\mathrm{orig}} + \lambda \Theta_0\right), \tag{24}$$

up to the transpose convention used to store output channels. This is the patchwise form of Eq. (3).

The correction uses no class labels. Its targets are the base model's internal Conv_2 outputs on the calibration images. A learned Conv_2 bias is included even though the base convolution is bias-free, allowing the augmented constant feature to absorb perturbation-induced mean shifts.

B.4 Chunked normal-equation accumulation

The bias-augmented correction dimension is

$$D_M = 1 + k_2^2 C_{\mathrm{in},2},$$

which is at most 4,609 in the evaluated ResNet blocks. The full unfolded design, however, contains $|S_m| H_2' W_2'$ rows and need not fit comfortably in GPU memory. We therefore accumulate the sufficient statistics in chunks:

$$H^{(m)} \leftarrow H^{(m)} + \overline{Y}_c^{(m)\top} \overline{Y}_c^{(m)}, \tag{25}$$

$$B^{(m)} \leftarrow B^{(m)} + \overline{Y}_c^{(m)\top} T_c^{\mathrm{orig}}. \tag{26}$$

After all chunks are processed, Eq. (24) is solved once. Chunk size changes only memory usage and not the resulting normal equations. It is configured by block to fit the available GPU memory; the selected final-stage block uses chunk size 64 in the reported joint-grid and headline runs.

B.5 Perturbation subspace

The random low-rank construction is identical to the fully connected case after flattening the Conv_1 kernel. We sample a Gaussian matrix in the d_1-dimensional kernel parameter space, orthonormalize its columns by QR factorization, and reshape sampled directions back to kernel form. The basis is shared across members and only the Gaussian coefficient vector varies. The selected CIFAR configuration uses $K = 20$ and $M = 50$.

B.6 Target-block selection and multi-block variants

The headline CIFAR-10 results use a single target block. We evaluate all six blocks in stages 2–4 in the joint block×scale×bootstrap search described in Appendix D.3; the first block of the final stage is selected by ID-validation NLL. The correction for this block has bias-augmented design dimension $1 + 9 \cdot 512 = 4{,}609$.

We also implement multi-block variants by applying the same local perturb-and-correct operation to multiple residual blocks in network order. Each selected block substitutes its member-specific perturbed Conv_1 and corrected Conv_2 parameters at inference, while all non-selected blocks retain the base parameters. These variants are used only for ablations and efficiency measurements in this paper; they are not used for the CIFAR headline comparison in Table 2.

B.7 Transformer feed-forward blocks

The ImageNet experiment applies P&C to the final feed-forward block of a pretrained ViT-B/16. Writing the tokenwise feed-forward map as

$$u = \phi(W_1 h + b_1), \qquad z = W_2 u + b_2,$$

with GELU nonlinearity ϕ, P&C perturbs W_1 and refits (W_2, b_2) to reproduce the base model's feed-forward output on ID calibration examples. The correction is therefore the same affine ridge problem as Eq. (1), with token representations playing the role of the design rows.

For the final transformer block, only the CLS row is needed in the correction fit. After this feed-forward block, the remaining operations before classification are tokenwise normalization followed by selection of the CLS representation and the classifier head; there is no later self-attention operation that can mix patch-token changes back into CLS. We therefore fit the reported ImageNet correction on final-block CLS representations. For an earlier transformer block, later attention can mix all token positions, so a corresponding correction would generally need to account for the relevant token rows rather than using CLS alone.

The ImageNet construction perturbs the first affine map of the final feed-forward block with a shared random rank-20 basis and uses ridge centered on the original second affine map. All attention parameters, LayerNorm parameters, and the classifier head remain fixed.

C MuJoCo Experimental Details

C.1 Compute and implementation details

The MuJoCo runs were executed sequentially on a single machine with an Intel i7-12700KF CPU and an NVIDIA TITAN X (Pascal) with 12 GB of GPU memory under WSL2. The models use JAX and a 4×200 ReLU MLP with diagonal-Gaussian mean and variance heads. Unless otherwise stated, P&C uses $K = 20$ perturbation directions and $M = 50$ members. The multi-block construction perturbs hidden layers 1 and 3 and corrects in the immediately following affine layers.

P&C is strictly post-training. The correction pool contains 4,096 expert-training transitions, and validation uses a disjoint held-out slice of the expert data. Near and Mid shifts are drawn from the corresponding Minari policy tiers when available; Far data are collected using uniform-random actions. No shifted data are used for model construction, hyperparameter selection, temperature fitting, or other post-hoc calibration.

C.2 Hyperparameter selection

Every method in Table 1 is selected independently per environment by ID-validation NLL. Final metrics are computed on disjoint ID and shifted evaluation data.

Shared protocol.

Architecture 4×200 MLP with ReLU activations and a heteroscedastic Gaussian output head.

Training $5{,}000$ optimizer steps, batch size 64.

Ensemble size Deep Ensemble uses $M = 5$. P&C uses $M = 50$; MC Dropout, SWAG, Subspace, and Laplace use 100 predictive samples.

Candidate grids.

MC Dropout Dropout probability in $\{0.05, 0.10, 0.20, 0.30, 0.50\}$; each value is trained separately.

SWAG Sampling scale in $\{0.1, 0.3, 1.0, 3.0, 10.0\}$.

Subspace Inference Temperature in $\{1, 10, 100, 1000, 10000\}$.

Laplace Prior precision in $\{10, 100, 1000, 10000, 100000\}$.

P&C Perturbation size in $\{8, 16, 32\}$, bootstrap fraction in $\{0.05, 0.10, 0.20, 0.30\}$, and ridge regularizer in $\{10^{-4}, 10^{-2}\}$, with $K = 20$ and $M = 50$ fixed.

All reported P&C corrections use ridge centered on the pretrained affine map.

C.3 Numerical validation of the residual analysis

The mechanism checks in Section 5.2 use Ant-v5, HalfCheetah-v5, and Hopper-v5 at model seed 0 with float64 instrumentation. For the exact residual identity, we evaluate both perturb–correct interfaces, members $\{0, 1, 3, 5\}$, $\lambda \in \{0, 10^{-2}, 1, 100\}$, and 1,024 points per available evaluation regime. Across correction, held-out ID, Near, Mid, and Far inputs, the direct post-correction residual and Eq. (7) agree to float64 precision: median relative errors are on the order of 10^{-13} and the

Table 4: P&C OOD-detection AUROC across graded MuJoCo shift tiers in the separate 27-seed diagnostic aggregation. This study uses a different run and aggregation protocol from the 10-seed validation-selected headline benchmark, so its absolute AUROC values are not directly comparable to the per-environment headline table. Mid-tier results are available for these four environments; other environments either lack a distinct Mid dataset or were evaluated only at the shared Far tier.

Environment	Near	Mid	Far
Ant-v5	0.862	0.915	0.998
HalfCheetah-v5	0.861	0.983	0.987
Hopper-v5	0.896	0.947	0.980
Humanoid-v5	0.570	0.522	0.343

observed maximum is at most 2.3×10^{-11}. The same stage-local identity also holds after an upstream perturb–correct stage has already modified the representations, with relative errors of approximately 10^{-13}–10^{-12} in the two-stage construction.

For the downstream bridge, we use the exact residual at the final corrected interface (hidden layer 3 corrected in layer 4). If z_0 is the base pre-activation entering the frozen output head F_{down}, the actual output change is $\Delta y = F_{\text{down}}(z_0 + r) - F_{\text{down}}(z_0)$. We compare this with $J_{\text{down}}(z_0)r$, where J_{down} is the Jacobian of the frozen downstream map at z_0. Across regimes and tested scales, the cosine similarity is 1.00 on Ant, 0.99–1.00 on Hopper, and 0.97–1.00 on HalfCheetah. This check concerns propagation of the *exact* corrected-layer residual through the downstream network; it is distinct from linearizing the correction itself in the perturbation.

To test that latter approximation, we evaluate Corollary 3 on the first perturb–correct interface over a logarithmic grid of perturbation norms, using three perturbation directions with $\lambda \in \{0, 1\}$. Taking relative error at most 10% as the quantitative-validity threshold, the derivative-at-zero approximation is accurate only for $\|\Delta W\|_F \lesssim 0.04$–$0.16$ across the three environments, while the experimental operating range is 5–50. Thus the applied perturbations are roughly 30–300$\times$ beyond the range in which the infinitesimal approximation is quantitatively accurate.

C.4 Sensitivity and direction diagnostics

The sensitivity study includes one-factor sweeps across all 11 environments and 27 seeds over perturbation scale, rank, bootstrap fraction, ridge, calibration-pool size, and layer scope. Each factor is varied around an environment-specific ID-selected anchor while all other settings are fixed. Rank is broadly stable through the tested central range; ridge has little predictive effect when correction is strongly overdetermined; and perturbation scale exhibits a clear ID-stability boundary. Near $n \approx p$, where $p = 201$ includes the bias coordinate, the correction interpolates calibration targets while held-out residuals peak. A small ridge suppresses this double-descent peak by one to three orders of magnitude in Ant, HalfCheetah, and Hopper phase diagrams. The full-bootstrap curve uses a 10,000-example pool and is therefore interpreted as mechanism evidence rather than a directly matched one-factor comparison.

Across five seeds on Ant, HalfCheetah, and Hopper, Random-P&C obtains higher Far-AUROC than Low-P&C in every seed. On Ant, mean Far-AUROC increases from 0.47 to 0.88; on Hopper, from 0.81 to 0.87. Low-P&C's ID-selected hidden perturbations shrink more strongly under shift and yield more aligned, lower-rank output changes, supporting random directions as a practical default.

C.5 Graded shift and layer scope

Across the 11-environment, 27-seed layer-scope study, macro-averaged Near/Mid/Far AUROC is $0.764/0.815/0.904$ for a first-block construction and $0.799/0.843/0.922$ for multi-block P&C.

C.6 SCOD protocol and results

SCOD is applied to the same frozen probabilistic MLP checkpoints and 4,096-point ID calibration sets. It constructs a randomized low-rank sketch of the dataset Fisher, with at most 100 eigenvectors and 604 sketch samples. Its hyperparameters are selected by ID-validation NLL before reading any OOD split. The comparison uses seeds $\{0, 10, 200\}$; the 27-seed P&C aggregation yields the same per-environment win ordering.

Table 5: Matched frozen-checkpoint comparison with SCOD on MuJoCo. Far-AUROC is averaged over environments in the first column; representative environments are shown separately. SCOD and the seed-matched P&C column each use seeds $\{0, 10, 200\}$.

Method	Mean	Ant	Hopper	Humanoid	Pusher
SCOD	0.512	0.110	0.028	0.267	0.995
P&C	**0.920**	**0.998**	**0.985**	**0.343**	**1.000**

Table 6: Measured ensemble-construction time on an NVIDIA TITAN X (Pascal), with $M = 50$. P&C time includes one base-model training run and post-hoc member construction. The compact size stores the base checkpoint and reconstructible P&C specification rather than 50 materialized members.

Environment	P&C (s)	Deep Ens. (s)	Ratio	P&C post-hoc (s)	Compact size (MB)
Ant-v5	18.4	1131.9	61.7×	1.75	0.76
HalfCheetah-v5	27.0	1321.9	49.0×	1.52	0.55
Hopper-v5	27.3	1279.3	46.8×	1.73	0.53

C.7 Efficiency protocol and results

Construction timing includes base-model training and all post-hoc work needed to materialize or sample $M = 50$ predictors. The matched Deep Ensemble estimate uses 50 independently trained networks. P&C compact storage records the base checkpoint and reconstructible perturbation/correction specification. In this MuJoCo implementation, both P&C and a matched Deep Ensemble require M full member forward evaluations, so the comparison does not claim an inference-time advantage.

C.8 Per-environment results

Table 7 gives the selected configurations. Table 8 reports ID RMSE, Far NLL, and Far AUROC by environment. Table 9 reports uncertainty–error Spearman correlations for the available tiers.

Table 7: Selected hyperparameter configuration per (env, method), chosen by lowest mean validation NLL across the seeds used in the main-paper tables. `n`: ensemble members; `dr`: dropout rate; `sws`: SWAG schedule weight scale; `prior`: Laplace prior precision; `T`: subspace temperature; `lreg`, `bf`, `ps`: P&C Tikhonov regulariser, per-member bootstrap fraction, and perturbation size.

Env	DE	MCD	SWAG	Lap.	Sub.	P&C
Ant-v5	n=5	dr=0.05	sws=1.0	prior=10000.0	T=1.0	lreg=0.0001, bf=0.1, ps=8.0
HalfCheetah-v5	n=5	dr=0.05	sws=0.3	prior=10000.0	T=1.0	lreg=0.0001, bf=0.3, ps=32.0
Hopper-v5	n=5	dr=0.05	sws=0.1	prior=10000.0	T=1.0	lreg=0.0001, bf=0.3, ps=32.0
Walker2d-v5	n=5	dr=0.05	sws=0.1	prior=10000.0	T=1.0	lreg=0.01, bf=0.3, ps=16.0
Swimmer-v5	n=5	dr=0.05	sws=0.1	prior=10000.0	T=1.0	lreg=0.0001, bf=0.05, ps=8.0
Humanoid-v5	n=5	default	sws=3.0	prior=10000.0	T=10.0	lreg=0.01, bf=0.3, ps=32.0
HumanoidStandup-v5	n=5	default	sws=3.0	prior=1000.0	T=10.0	lreg=0.01, bf=0.2, ps=8.0
Reacher-v5	n=5	dr=0.05	sws=0.1	prior=10000.0	T=1.0	lreg=0.0001, bf=0.05, ps=8.0
Pusher-v5	n=5	dr=0.05	sws=1.0	prior=10000.0	T=10.0	lreg=0.0001, bf=0.1, ps=8.0
InvertedPendulum-v5	n=5	dr=0.05	sws=0.1	prior=10000.0	T=1.0	lreg=0.0001, bf=0.05, ps=16.0
InvertedDoublePendulum-v5	n=5	dr=0.2	sws=0.3	prior=10000.0	T=10.0	lreg=0.0001, bf=0.05, ps=16.0

Table 8: Per-environment headline metrics on MuJoCo. Cells report mean ± stddev across seeds for the validation-NLL-selected configuration. Header arrows indicate the preferred direction. Within each environment, the best value(s) in each metric column are shown in **bold**.

Method	ID RMSE ↓	Far NLL ↓	Far AUROC ↑
Ant-v5 ($n = 10$ seeds)			
Deep Ensemble	0.601 ± 0.022	14.361 ± 10.155	0.579 ± 0.075
MC Dropout	0.571 ± 0.076	1374.124 ± 980.958	0.379 ± 0.120
SWAG	0.674 ± 0.049	**1.186 ± 0.358**	0.476 ± 0.152
Laplace	0.590 ± 0.093	6.438 ± 3.907	0.401 ± 0.167
Subspace	0.552 ± 0.013	30.496 ± 24.740	0.302 ± 0.106

Continued on next page

Table 8: Per-environment headline metrics on MuJoCo (continued)

Method	ID RMSE ↓	Far NLL ↓	Far AUROC ↑
P&C	**0.289 ± 0.151**	1.544 ± 0.348	**0.993 ± 0.003**
HalfCheetah-v5 ($n = 10$ seeds)			
Deep Ensemble	1.323 ± 0.073	9.665 ± 5.843	0.969 ± 0.017
MC Dropout	1.463 ± 0.103	6.087 ± 4.379	0.939 ± 0.026
SWAG	1.493 ± 0.083	7.663 ± 5.000	0.964 ± 0.022
Laplace	**1.291 ± 0.088**	5.568 ± 3.110	0.945 ± 0.024
Subspace	1.641 ± 0.078	19.917 ± 34.424	0.955 ± 0.032
P&C	1.779 ± 0.800	**3.735 ± 0.456**	**0.986 ± 0.010**
Hopper-v5 ($n = 10$ seeds)			
Deep Ensemble	**0.167 ± 0.010**	2.637 ± 0.467	0.903 ± 0.019
MC Dropout	0.210 ± 0.018	2.115 ± 0.683	0.624 ± 0.046
SWAG	0.180 ± 0.009	4.579 ± 2.421	0.792 ± 0.048
Laplace	0.175 ± 0.016	2.757 ± 0.823	0.641 ± 0.081
Subspace	0.194 ± 0.008	5.443 ± 2.539	0.782 ± 0.055
P&C	0.219 ± 0.012	**0.825 ± 0.080**	**0.979 ± 0.007**
Walker2d-v5 ($n = 10$ seeds)			
Deep Ensemble	0.591 ± 0.023	9.450 ± 2.146	0.972 ± 0.005
MC Dropout	0.566 ± 0.026	15.869 ± 3.836	0.945 ± 0.013
SWAG	0.661 ± 0.016	28.842 ± 10.908	0.909 ± 0.030
Laplace	0.579 ± 0.019	9.913 ± 2.415	0.897 ± 0.023
Subspace	0.729 ± 0.016	36.106 ± 15.215	0.904 ± 0.030
P&C	**0.551 ± 0.019**	**1.839 ± 0.029**	**0.991 ± 0.001**
Swimmer-v5 ($n = 10$ seeds)			
Deep Ensemble	0.027 ± 0.012	3.498 ± 0.492	**0.998 ± 0.005**
MC Dropout	0.058 ± 0.004	4.829 ± 0.645	0.992 ± 0.002
SWAG	0.056 ± 0.086	38.232 ± 20.216	0.923 ± 0.076
Laplace	0.035 ± 0.014	11.051 ± 2.425	0.988 ± 0.008
Subspace	**0.026 ± 0.006**	76.561 ± 42.428	0.956 ± 0.024
P&C	0.051 ± 0.003	**1.696 ± 0.439**	0.990 ± 0.003
Humanoid-v5 ($n = 10$ seeds)			
Deep Ensemble	51.815 ± 6.334	4.234 ± 0.235	0.285 ± 0.053
MC Dropout	44.184 ± 2.090	3.791 ± 0.385	0.288 ± 0.055
SWAG	46.966 ± 8.429	3.745 ± 0.288	0.187 ± 0.027
Laplace	64.620 ± 53.173	4.298 ± 1.239	0.251 ± 0.072
Subspace	43.199 ± 2.519	4.377 ± 0.534	0.173 ± 0.024
P&C	**27.215 ± 0.586**	**3.274 ± 0.090**	**0.837 ± 0.029**
HumanoidStandup-v5 ($n = 10$ seeds)			
Deep Ensemble	33.783 ± 7.675	7.065 ± 0.546	0.543 ± 0.113
MC Dropout	23.264 ± 3.640	6.258 ± 1.261	0.726 ± 0.072
SWAG	23.211 ± 3.301	5.836 ± 0.638	0.575 ± 0.143
Laplace	39.908 ± 26.810	**5.614 ± 0.871**	0.650 ± 0.122
Subspace	**21.107 ± 1.290**	25.971 ± 22.086	0.577 ± 0.135
P&C	25.13 ± 5.226	5.757 ± 0.153	**0.977 ± 0.017**
Reacher-v5 ($n = 10$ seeds)			
Deep Ensemble	**0.087 ± 0.009**	2.587 ± 0.423	0.971 ± 0.004
MC Dropout	0.111 ± 0.009	7.940 ± 0.930	0.929 ± 0.004
SWAG	0.089 ± 0.011	190.001 ± 53.046	0.816 ± 0.055
Laplace	0.088 ± 0.011	1.974 ± 0.404	0.926 ± 0.008
Subspace	0.090 ± 0.009	1015.748 ± 653.260	0.771 ± 0.058
P&C	0.120 ± 0.021	**0.917 ± 0.106**	**0.983 ± 0.003**
Pusher-v5 ($n = 10$ seeds)			
Deep Ensemble	0.054 ± 0.009	5.899 ± 1.033	0.997 ± 0.006
MC Dropout	0.057 ± 0.001	20.634 ± 4.510	0.972 ± 0.005
SWAG	0.094 ± 0.051	28.527 ± 18.351	0.625 ± 0.309
Laplace	0.054 ± 0.009	17.909 ± 2.376	0.900 ± 0.076
Subspace	**0.053 ± 0.002**	2268.085 ± 1140.032	0.253 ± 0.091

Continued on next page

Table 8: Per-environment headline metrics on MuJoCo (continued)

Method	ID RMSE ↓	Far NLL ↓	Far AUROC ↑
P&C	0.054 ± 0.002	$\mathbf{2.012 \pm 0.227}$	$\mathbf{1.000 \pm 0.000}$
InvertedPendulum-v5 ($n = 10$ seeds)			
Deep Ensemble	0.017 ± 0.008	-0.059 ± 0.811	0.832 ± 0.096
MC Dropout	0.010 ± 0.002	-0.508 ± 0.452	0.934 ± 0.010
SWAG	0.052 ± 0.048	94.165 ± 65.723	0.488 ± 0.061
Laplace	0.014 ± 0.014	1.178 ± 1.002	0.867 ± 0.148
Subspace	0.015 ± 0.004	316.721 ± 181.217	0.559 ± 0.086
P&C	$\mathbf{0.006 \pm 0.002}$	$\mathbf{-1.334 \pm 0.447}$	$\mathbf{0.956 \pm 0.006}$
InvertedDoublePendulum-v5 ($n = 10$ seeds)			
Deep Ensemble	0.028 ± 0.013	$\mathbf{4.230 \pm 1.265}$	0.994 ± 0.004
MC Dropout	0.030 ± 0.006	108.931 ± 75.299	$\mathbf{0.997 \pm 0.001}$
SWAG	0.095 ± 0.057	124.220 ± 54.583	0.398 ± 0.289
Laplace	0.023 ± 0.011	18.250 ± 4.784	0.951 ± 0.092
Subspace	0.025 ± 0.005	2933.610 ± 2712.973	0.650 ± 0.215
P&C	$\mathbf{0.020 \pm 0.004}$	4.942 ± 0.542	0.995 ± 0.001

Table 9: Spearman rank correlation between predicted variance and squared error, computed per seed and reported as mean ± stddev across seeds. Header arrows indicate that higher is better. Within each environment, the best value(s) in each split column are shown in **bold**. Negative correlations are italicized to indicate anti-informative uncertainty signals.

Method	ID ↑	Near ↑	Mid ↑	Far ↑
Ant-v5				
Deep Ensemble	0.32 ± 0.04	0.23 ± 0.04	0.08 ± 0.05	0.09 ± 0.04
MC Dropout	0.38 ± 0.15	0.23 ± 0.13	-0.02 ± 0.11	0.06 ± 0.07
SWAG	0.46 ± 0.11	0.25 ± 0.07	-0.08 ± 0.06	0.07 ± 0.06
Laplace	0.44 ± 0.09	0.28 ± 0.08	0.04 ± 0.11	0.20 ± 0.12
Subspace	0.30 ± 0.04	0.21 ± 0.06	0.01 ± 0.05	0.22 ± 0.09
P&C	0.72 ± 0.05	0.53 ± 0.09	0.59 ± 0.08	0.31 ± 0.13
HalfCheetah-v5				
Deep Ensemble	0.39 ± 0.04	0.61 ± 0.06	-0.09 ± 0.09	0.10 ± 0.04
MC Dropout	0.39 ± 0.08	0.56 ± 0.09	-0.12 ± 0.14	0.08 ± 0.05
SWAG	0.40 ± 0.05	0.42 ± 0.10	-0.09 ± 0.15	0.11 ± 0.05
Laplace	0.47 ± 0.04	0.57 ± 0.09	-0.12 ± 0.19	0.16 ± 0.07
Subspace	0.37 ± 0.06	0.33 ± 0.10	-0.07 ± 0.13	0.13 ± 0.05
P&C	0.44 ± 0.08	0.78 ± 0.05	0.69 ± 0.14	0.85 ± 0.09
Hopper-v5				
Deep Ensemble	0.65 ± 0.02	0.64 ± 0.02	0.59 ± 0.04	0.52 ± 0.05
MC Dropout	0.61 ± 0.03	0.35 ± 0.06	0.37 ± 0.10	0.55 ± 0.07
SWAG	0.67 ± 0.03	0.61 ± 0.06	0.05 ± 0.18	0.24 ± 0.13
Laplace	0.66 ± 0.06	0.60 ± 0.05	0.49 ± 0.09	0.38 ± 0.11
Subspace	0.64 ± 0.03	0.60 ± 0.06	0.09 ± 0.19	0.23 ± 0.12
P&C	0.49 ± 0.07	0.61 ± 0.05	0.60 ± 0.07	0.73 ± 0.07
Walker2d-v5				
Deep Ensemble	0.46 ± 0.01	0.45 ± 0.03	0.33 ± 0.04	0.26 ± 0.03
MC Dropout	0.51 ± 0.03	0.45 ± 0.05	0.35 ± 0.08	0.09 ± 0.08
SWAG	0.47 ± 0.02	0.30 ± 0.08	0.34 ± 0.05	0.12 ± 0.07
Laplace	0.51 ± 0.03	0.39 ± 0.05	0.35 ± 0.07	0.20 ± 0.08
Subspace	0.43 ± 0.03	0.26 ± 0.07	0.33 ± 0.04	0.09 ± 0.05
P&C	0.51 ± 0.02	0.50 ± 0.05	0.36 ± 0.02	0.36 ± 0.04
Swimmer-v5				
Deep Ensemble	0.53 ± 0.14	0.76 ± 0.05	–	0.48 ± 0.05
MC Dropout	0.46 ± 0.09	0.45 ± 0.08	–	0.26 ± 0.12
SWAG	0.59 ± 0.16	-0.19 ± 0.23	–	-0.33 ± 0.06
Laplace	0.41 ± 0.25	0.39 ± 0.13	–	-0.21 ± 0.08
Subspace	0.52 ± 0.06	-0.20 ± 0.09	–	-0.29 ± 0.07
P&C	0.65 ± 0.09	0.80 ± 0.04	–	0.64 ± 0.05
Humanoid-v5				
Deep Ensemble	0.91 ± 0.01	0.64 ± 0.03	0.42 ± 0.10	0.76 ± 0.03
MC Dropout	0.86 ± 0.03	0.56 ± 0.06	0.22 ± 0.05	0.62 ± 0.04
SWAG	0.88 ± 0.04	0.57 ± 0.07	0.28 ± 0.15	0.69 ± 0.07
Laplace	0.92 ± 0.02	0.65 ± 0.11	0.41 ± 0.24	0.76 ± 0.07
Subspace	0.84 ± 0.05	0.51 ± 0.07	0.17 ± 0.08	0.64 ± 0.04
P&C	0.83 ± 0.02	0.65 ± 0.03	0.42 ± 0.05	0.71 ± 0.03
HumanoidStandup-v5				
Deep Ensemble	0.65 ± 0.18	0.71 ± 0.12	0.84 ± 0.08	0.73 ± 0.11
MC Dropout	0.18 ± 0.14	0.68 ± 0.16	0.82 ± 0.09	0.67 ± 0.12
SWAG	0.30 ± 0.17	0.60 ± 0.20	0.78 ± 0.12	0.55 ± 0.12
Laplace	0.61 ± 0.26	0.79 ± 0.17	0.90 ± 0.07	0.79 ± 0.14
Subspace	0.18 ± 0.11	0.53 ± 0.22	0.74 ± 0.11	0.51 ± 0.10
P&C	0.81 ± 0.07	0.99 ± 0.01	0.99 ± 0.01	0.99 ± 0.01
Reacher-v5				
Deep Ensemble	0.60 ± 0.08	0.48 ± 0.08	–	0.85 ± 0.02
MC Dropout	0.63 ± 0.05	0.55 ± 0.07	–	0.73 ± 0.03
SWAG	0.59 ± 0.06	0.39 ± 0.12	–	0.43 ± 0.20

Continued on next page

Table 9: Spearman rank correlation between predicted variance and squared error (continued)

Method	ID ↑	Near ↑	Mid ↑	Far ↑
Laplace	0.58 ± 0.13	0.43 ± 0.11	–	0.79 ± 0.02
Subspace	0.58 ± 0.04	0.46 ± 0.06	–	0.34 ± 0.20
P&C	0.69 ± 0.06	0.63 ± 0.08	–	0.70 ± 0.04
Pusher-v5				
Deep Ensemble	0.59 ± 0.06	0.74 ± 0.08	–	0.78 ± 0.02
MC Dropout	0.65 ± 0.03	0.70 ± 0.02	–	0.45 ± 0.05
SWAG	0.33 ± 0.15	0.14 ± 0.33	–	0.24 ± 0.35
Laplace	0.62 ± 0.05	0.58 ± 0.19	–	0.62 ± 0.09
Subspace	0.58 ± 0.03	0.31 ± 0.09	–	-0.47 ± 0.09
P&C	0.61 ± 0.05	0.85 ± 0.04	–	0.77 ± 0.02
InvertedPendulum-v5				
Deep Ensemble	0.76 ± 0.14	0.76 ± 0.10	–	0.69 ± 0.08
MC Dropout	0.50 ± 0.10	0.59 ± 0.06	–	0.54 ± 0.08
SWAG	0.61 ± 0.20	-0.24 ± 0.22	–	-0.57 ± 0.25
Laplace	0.46 ± 0.22	0.41 ± 0.34	–	0.49 ± 0.41
Subspace	0.46 ± 0.09	-0.01 ± 0.13	–	-0.58 ± 0.22
P&C	0.41 ± 0.18	0.90 ± 0.02	–	0.86 ± 0.03
InvertedDoublePendulum-v5				
Deep Ensemble	0.57 ± 0.17	0.44 ± 0.12	–	0.88 ± 0.02
MC Dropout	0.35 ± 0.10	0.45 ± 0.10	–	0.44 ± 0.08
SWAG	0.31 ± 0.45	0.12 ± 0.33	–	-0.52 ± 0.16
Laplace	0.36 ± 0.21	0.27 ± 0.18	–	0.68 ± 0.17
Subspace	0.18 ± 0.10	0.10 ± 0.13	–	-0.39 ± 0.25
P&C	0.60 ± 0.15	0.78 ± 0.07	–	0.85 ± 0.03

D CIFAR-10 protocol, selection, and sensitivity

D.1 Data splits and selection protocol

CIFAR-10 contains 50,000 training and 10,000 test images. We reserve 10% of the training set (5,000 images) as an ID validation split using a fixed seed and use the remaining 45,000 images for base-model training. Hyperparameters are selected by ID-validation NLL after post-hoc temperature scaling; no OOD data are used for construction, hyperparameter selection, or temperature fitting.

P&C uses a 1,024-example correction pool drawn without replacement from the 45,000-image training-fit split. For a given checkpoint, the same base correction pool is reused across hyperparameter configurations. When bootstrapping is enabled, each member resamples within this pool. The affine correction uses only intermediate activations and the base model's pre-activation targets; class labels are not used by the correction solve. Validation, CIFAR-10 test, and OOD examples never enter the correction fit.

For classifier inference, one scalar temperature is fitted per configuration on the 5,000-example ID-validation split. The same temperature is applied to every member's logits before softmax, member probabilities are averaged, and predictive entropy is used as the P&C OOD score. FPR95 is the false-positive rate at the first threshold reaching at least 95% true-positive rate. Near and Far metrics are arithmetic means over the datasets in their respective OpenOOD groups.

The shared PreAct ResNet-18 training recipe uses SGD with Nesterov momentum 0.9, initial learning rate 0.1 with cosine decay and a five-epoch warmup, weight decay 5×10^{-4}, batch size 128, 300 epochs, flip/crop plus cutout(8) augmentation, and no label smoothing.

D.2 Hyperparameter grids

Table 10 lists the principal selection grids. Secondary settings not shown were fixed to their implementation or literature defaults.

Table 10: CIFAR-10 hyperparameter grids. Bold marks the selected P&C values. The P&C target block, scale, and bootstrap fraction were evaluated jointly as a complete Cartesian product; rank and ridge were studied separately in one-factor sensitivity sweeps.

Method	Hyperparameter	Values
MC Dropout	dropout rate	$\{0.1, \mathbf{0.2}, 0.3, 0.5\}$
LLLA	prior precision	$\{0.01, 0.1, 1, \mathbf{10}, 100, 1000\}$
SWAG	start epoch	$\{160, \mathbf{240}\}$
Epinet	prior scale	$\{0.5, 1, \mathbf{3}\}$
Deep Ensemble	number of models	5
SCOD	calibration size / sketch rank	$1024/10$
P&C	target (stage, block)	$\{1, 2, 3\} \times \{0, 1\}$; $(\mathbf{3}, \mathbf{0})$
	perturbation scale σ	$\{\mathbf{25}, 50, 100\}$
	bootstrap fraction	$\{\mathbf{0.05}, 0.10, 0.20\}$
	perturbation rank K	20
	ridge λ	10^{-3}
	ensemble size / correction pool	$50/1024$

D.3 Joint P&C selection

To test interactions among the three principal P&C selection dimensions, we evaluate all six target blocks, three perturbation scales, and three bootstrap fractions: $6 \times 3 \times 3 = 54$ configurations per checkpoint and 162 runs across checkpoints/seeds $\{0, 1, 2\}$. All configurations use $K = 20$, $M = 50$, $\lambda = 10^{-3}$, a 1,024-example correction pool, random directions, frozen batch-normalization statistics, and ridge centered on the original corrected convolution.

Selection minimizes mean temperature-scaled ID-validation NLL across the three checkpoints. The global optimum within this grid is stage 3, block 0 (the first block of the final ResNet stage), $\sigma = 25$, and bootstrap fraction 0.05, with validation NLL 0.1305 ± 0.0049. This configuration ranks first among all 54 combinations. Coordinate-wise selection in either forward or reverse order reaches the

Table 11: CIFAR-10 P&C sensitivity around the ID-selected configuration. Each entry aggregates checkpoints/seeds $\{0, 1, 2\}$. Scale, rank, ridge, and target block are one-factor sweeps; bootstrap values are from the joint block×scale×bootstrap grid at the selected block and scale.

Factor	Sweep	Anchor	Main observation
Scale σ	5 values, 6.25–100	25	Near AUROC peaks at 91.55; ID accuracy collapses for $\sigma \geq 50$
Rank K	5 values, 1–40	20	Near AUROC is 91.40–91.55 for $K \geq 5$
Ridge λ	6 values, 0–1	10^{-3}	$\lambda = 0$ is singular/unstable; $\lambda \geq 10^{-3}$ is nearly flat
Target block	early, middle, late	late (s3b0)	val NLL $0.147/0.151/0.130$; all three remain viable
Bootstrap frac.	.05, .10, .20	.05	val NLL $0.1305/0.1357/0.1452$ at s3b0, $\sigma = 25$

same configuration, so the selected result is not an artifact of the coordinate-wise search procedure. OOD datasets are loaded only after this configuration is fixed.

D.4 One-factor sensitivity

The scale sweep shows the clearest stability boundary: performance improves through $\sigma = 25$ but large perturbations cross the ID-preservation regime, with mean ID accuracy falling to roughly 35% at $\sigma = 50$ and 10% at $\sigma = 100$. Rank is much less sensitive: Near AUROC is essentially flat from $K = 5$ through $K = 40$. Ridge has a threshold effect rather than being irrelevant. The unregularized solve is singular or numerically unstable, $\lambda = 10^{-4}$ is degraded, and $\lambda \in [10^{-3}, 1]$ produces nearly identical predictive performance.

Early, middle, and late target blocks all yield viable predictors at the anchor settings. Their mean accuracies span 94.76–95.59%, Near AUROC 89.34–91.55, and Far AUROC 92.01–95.09; the final-stage first block is selected because it has the lowest ID-validation NLL (0.130 versus 0.147 and 0.151 for the representative early and middle blocks).

Bootstrap fraction is conditioning-sensitive. In the completed joint grid at the selected block and scale, fractions 0.05, 0.10, and 0.20 yield validation NLL 0.1305, 0.1357, and 0.1452, respectively. A separate refined one-factor harness produces an instability at 0.10 that does not reproduce in the joint grid, so we do not interpret that isolated collapse as a robust double-descent effect. A calibration-size sweep likewise contains a reproducible numerical/conditioning failure at 2,048 examples while neighboring sizes remain healthy, reinforcing that correction conditioning, rather than sample count alone, is an important practical constraint.

D.5 Matched SCOD comparison

SCOD and P&C use the same three PreAct ResNet-18 checkpoints and matched 1,024-example ID calibration pools. The SCOD construction uses sketch rank 10 and is selected without OOD data. Table 2 compares SCOD with the same P&C configuration selected above. SCOD obtains Near/Far AUROC $89.69/92.56$, while P&C obtains $91.55/95.09$; corresponding FPR95 values are $39.40/21.41$ for SCOD and $33.08/18.15$ for P&C.

D.6 Efficiency, storage, and memory

The measured base-model training time is 8,010 seconds. P&C's post-training construction was not instrumented as a standalone timer in the anchor run; subtracting the measured OpenOOD evaluation from the combined build-and-evaluate run gives a conservative upper bound of approximately 100 seconds. We therefore report this quantity explicitly as an upper bound rather than a direct measurement. Deep-Ensemble construction time in Table 12 is derived by multiplying the measured base-training time by the number of independently trained members.

At matched $M = 50$, P&C and the Deep Ensemble have essentially the same measured inference latency (7.53 versus 7.26 ms/example), as expected because both perform 50 member forward passes. P&C therefore has no matched-member inference-time advantage. Its efficiency advantage relative to a 50-member Deep Ensemble is in avoiding 49 additional training runs and in persistent representation: the benchmark implementation reconstructs P&C from one 42.7-MiB base checkpoint, while 50 independent checkpoints require approximately 2.13 GiB. The memory measurements show

Table 12: CIFAR-10 construction and matched-member inference accounting. Inference uses $5{,}000$ images, batch size 256, one synchronized warm run on seed 0. P&C construction is a derived upper bound beyond one measured $8{,}010$-s base training run; Deep Ensemble training is derived as $50\times$ the same base-training time. SWAG and LLLA report measured post-training construction.

Method	Training requirement	Post-training construction	Inference (ms/example)
P&C ($M = 50$)	1 base (8,010 s)	$\leq$ 100 s (upper bound)	7.53
Deep Ensemble ($M = 50$)	50 bases ($\approx$ 400,500 s)	–	7.26
SWAG ($M = 50$)	1 trajectory fit	55.8 ± 5.1 s	7.33
LLLA ($M = 50$)	1 base	15.56 ± 0.30 s	7.68

Table 13: CIFAR-10 storage and GPU-memory accounting. GPU resident and peak memory are measured for seed 0 with JAX preallocation disabled. Persistent P&C storage is the 42.7-MiB base checkpoint because members are reconstructed; materializing the 50 compact convolutional corrections would require approximately 450 MiB (derived). Deep-Ensemble storage is derived as $50\times$ the measured base-checkpoint size.

Method	Persistent representation (MiB)	Resident GPU (MiB)	Peak GPU (MiB)
P&C ($M = 50$)	42.7 reconstructed ($\sim$ 450 materialized)	1,063	4,332
Deep Ensemble ($M = 50$)	2,133 (derived)	2,250	4,072
SWAG ($M = 50$)	980	1,071	5,360
LLLA ($M = 50$)	–	245	2,097

that this construction advantage does not imply uniformly lower peak GPU memory; P&C peaks at 4.33 GiB in the measured run versus 4.07 GiB for the matched Deep Ensemble, while SWAG and LLLA have different storage and memory tradeoffs.

E ImageNet-1K ViT experiment

We evaluate P&C on a pretrained ViT-B/16 ImageNet-1K checkpoint with 86.6M parameters. The construction targets the final transformer feed-forward block: P&C perturbs the first affine map of the FFN, applies the fixed GELU nonlinearity, and refits the second affine map. As described in Appendix B.7, the correction is fit on final-block CLS representations rather than patch-token rows.

We partition ImageNet training examples into disjoint pools for correction (32,768 images), ID selection (8,192), and temperature fitting (8,192). The official 50,000-image validation set is left untouched until final evaluation. P&C uses perturbation rank $K = 20$ and final ensemble size $M = 20$. Member probabilities are temperature-scaled with a shared scalar temperature $T = 0.700$ fit on the ID temperature pool, then averaged; predictive entropy is used for OOD ranking. Near OOD comprises SSB-hard and NINCO, while Far OOD comprises iNaturalist, Textures, and OpenImage-O.

E.1 ID-preservation selection

The perturbation scale is selected by asking how large a perturbation the corrected model can support while preserving ID accuracy. The primary operating point uses a fixed 0.50 percentage-point top-1 preservation budget. For each candidate perturbation magnitude $r = \|\Delta W_1\|_F / \|W_1\|_F$, ridge strength is selected using ID data only, and preservation is checked using a one-sided 95% lower confidence bound from a paired bootstrap over the ID-selection examples. The matched uncorrected ensemble is an ablation and does not enter selection.

This procedure selects $r = 2.0$ and $\lambda = 1000$ for the primary budget. The reported λ corresponds to the summed least-squares objective in Eq. (1); with 32,768 CLS correction rows, its mean-objective equivalent is approximately 0.0305. The corrected ensemble passes through $r = 2.0$ and fails the primary preservation criterion by $r = 2.125$. For sensitivity, 0.25- and 1.00-percentage-point budgets select $r = 1.25$ and $r = 3.0$, respectively, with the same selected ridge. These operating points are fixed before the follow-up OOD evaluation. The preservation-budget protocol itself was developed after an initial feasibility study that had inspected OOD behavior at several perturbation scales; we therefore treat the follow-up as an ID-selected diagnostic rather than a preregistered OOD study.

On the official ImageNet validation set, the base checkpoint obtains 81.068% top-1 accuracy, NLL 0.8482, and ECE 0.0913. The primary P&C ensemble obtains 80.830% top-1 accuracy, NLL 0.8045, and ECE 0.0542. Thus the selected ensemble remains within the preservation budget while improving both NLL and ECE under the shared temperature. The same $r = 2$ perturbations without correction obtain 80.619% accuracy and NLL 0.9048.

The role of correction becomes more pronounced as perturbation magnitude increases. At $r = 1$, corrected and uncorrected ID accuracy are nearly indistinguishable; at $r = 2$ the corrected model begins to preserve accuracy better, and at $r = 4$ the corrected ensemble loses 0.960 percentage points while the uncorrected ensemble loses 10.754 points. This scale dependence is consistent with the intended role of the affine repair: small perturbations require little correction, whereas larger perturbations expose a regime in which the repair is necessary to retain the base model's ID behavior.

E.2 OOD detection and matched ablation

Table 3 reports the frozen-checkpoint comparison. At the primary operating point, P&C obtains Near/Far AUROC $76.49/87.89$, compared with $72.53/84.40$ for the identical perturbations without correction and $73.52/86.04$ for MSP. Thus correction improves the matched perturbation ensemble by 3.96 Near-AUROC points and 3.49 Far-AUROC points; without correction, the $r = 2$ perturbation ensemble is worse than MSP on both aggregates.

LLLA-Kron obtains Near/Far AUROC $72.95/87.42$, while a restricted linear-scope SCOD comparison obtains $75.15/88.48$. P&C is therefore stronger than LLLA-Kron on Near OOD and slightly stronger on Far, and is stronger than SCOD-linear on Near while weaker on Far. A class-conditional Mahalanobis detector on the final normalized CLS representation remains stronger for pure OOD ranking, with Near/Far AUROC $78.82/92.55$. This detector leaves the predictive distribution unchanged, whereas P&C constructs an ensemble predictive distribution. We therefore interpret the comparison as evidence that direct representation-space unsupportedness can be a stronger OOD score on this benchmark, rather than as evidence that the two methods estimate the same uncertainty object.

Across the strict, primary, and relaxed preservation budgets, P&C Near/Far AUROC is $75.59/87.38$, $76.49/87.89$, and $77.27/88.06$, respectively. OOD ranking therefore improves as more of the allowed ID-preservation budget is used over the tested range. We retain the 0.50-percentage-point budget as the primary result rather than selecting the relaxed configuration from OOD performance.

E.3 Representation geometry

The stronger Mahalanobis result is not explained simply by the rank of the P&C perturbation subspace. Holding the same preservation protocol fixed, Near-OOD AUROC is 76.53, 76.49, 76.51, and 76.41 for $K \in \{5, 20, 40, 80\}$. A random 20-dimensional projection of the final CLS representation also gives substantially weaker Mahalanobis performance than the full representation.

Instead, the methods emphasize different parts of the final representation geometry. Mahalanobis separation is strongest in low-variance ID directions: using the lowest-variance 96 covariance modes gives Near-OOD AUROC 81.81, compared with 68.32 for the highest-variance 96 modes. P&C's induced final-CLS response has the opposite spectral weighting. Across covariance eigenmodes, response energy has Spearman correlation 0.993 with the corresponding ID eigenvalue; 26.7% of response energy lies in the highest-variance 96 modes and 4.69% in the lowest-variance 96. This provides a concrete explanation for why representation distance can flag inputs that remain comparatively stable across the P&C members.

E.4 Construction and inference cost

For the primary $M = 20$ construction, the complete post-hoc procedure takes 284.7 seconds from a loaded base checkpoint and prepared data manifests. Of this, 276.3 seconds is the one-time feature-cache pass; constructing the member-specific perturbations and corrections after caching takes 8.4 seconds. Peak GPU memory is 1.49 GiB. The base checkpoint plus compact P&C representation occupies approximately 510 MiB, compared with a derived 6.61 GiB for 20 complete ViT checkpoints.

The final-block target also allows almost the entire transformer prefix to be shared at inference. On the same GPU, the base ViT evaluates at 100.4 images/s and the cached-prefix $M = 20$ P&C ensemble

at 98.4 images/s. The member-dependent tail still scales as $O(M)$; the small measured overhead arises because only the final feed-forward block and classifier-side computation must be evaluated separately for each member.

F Additional DistilBERT transfer experiment

We additionally apply P&C inside the final feed-forward block of a fine-tuned 67M-parameter DistilBERT Banking77 classifier. This setting introduces self-attention, residual connections, LayerNorm, and categorical text prediction without retraining the checkpoint.

Table 14: Banking77 ID prediction and CLINC-OOS detection with a 67M-parameter DistilBERT checkpoint. All methods use the same fine-tuned checkpoint. Internal-FFN P&C and Uncorrected perturbation are averaged over five post-hoc construction seeds $\{0, 10, 42, 123, 2026\}$; MC Dropout is averaged over the same five stochastic inference seeds. Energy is deterministic for the fixed checkpoint.

Method	ID Acc.	ID NLL	ID ECE	Near AUROC	Cross AUROC	Far AUROC
Energy	0.925	0.303	0.032	**0.929**	**0.980**	**0.990**
MC Dropout	0.925	**0.283**	**0.013**	0.917	0.969	0.982
Uncorrected perturb.	0.921	1.079	0.525	0.885	0.972	0.989
Internal-FFN P&C	0.925	0.301	0.031	0.907	0.967	0.981

Internal-FFN P&C preserves 92.5% ID accuracy with NLL 0.301 and ECE 0.031. Applying the same perturbations without affine correction retains 92.1% accuracy but degrades NLL to 1.079 and ECE to 0.525, showing that the repair remains important in this architecture. On CLINC-OOS, P&C is competitive with MC Dropout, while Energy remains the strongest OOD detector in this setting. We use this experiment as an additional test of whether the same post-hoc construction can operate inside a different model family rather than as a comprehensive language OOD benchmark.

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: Claims are backed by proofs in a theory section and the appendix, as well as empirical backing in an experiments section.

   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: Assumptions made in the theory are laid out explicitly. The scope of hyperparameter tuning is discussed in the appendix. Computational requirements are similar to similar methods and doesn't scale with dataset size. A dedicated limitations section is elided due to the page limit.

   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**

Confidential reviewer copy. This manuscript is submitted to the 40th Conference on Neural Information Processing Systems (NeurIPS 2026). Unauthorized sharing, redistribution, or disclosure is strictly prohibited.

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

Answer: [Yes]

Justification: Assumptions are included and full proofs are 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: The algorithm is detailed in the paper body. Experimental details used to collect empirical data are included in the 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.

Confidential reviewer copy. This manuscript is submitted to the 40th Conference on Neural Information Processing Systems (NeurIPS 2026). Unauthorized sharing, redistribution, or disclosure is strictly prohibited.

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]

   Justification: All data is available as datasets online or is easily collected with publicly assessible tools (in the case of MuJoCo Far data). Code is included in the supplementary materials.

   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: Details for each of the MuJoCo and CIFAR settings can be found in the corresponding appendix.

   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: Tables include standard deviations and the MuJoCo table indicates statistical significance with the Wilcoxon-Holm test.

   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.

Confidential reviewer copy. This manuscript is submitted to the 40th Conference on Neural Information Processing Systems (NeurIPS 2026). Unauthorized sharing, redistribution, or disclosure is strictly prohibited.

- 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: Computational details can be found in appendix C.1.

   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: The research fully conforms to the 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: The authors do not forsee any harms arising from the proposed algorithm being used as intended. Gross misuse could result in overconfidence in the outputs of a model, though this has no immediate or obvious social impact.

    Guidelines:

Confidential reviewer copy. This manuscript is submitted to the 40th Conference on Neural Information Processing Systems (NeurIPS 2026). Unauthorized sharing, redistribution, or disclosure is strictly prohibited.

- 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: We only describe a novel algorithm with analysis and experiments. The authors don't forsee any potential for direct misuse.

    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: Existing assets and licenses. We use CIFAR-10 for image classification benchmarking and follow the OpenOOD v1.5 CIFAR-10 OOD detection protocol, including the benchmark's near- and far-OOD datasets. We use Gymnasium MuJoCo environments, backed by the MuJoCo physics engine, and the pre-collected MuJoCo Minari datasets. OpenOOD is MIT-licensed, Gymnasium is MIT-licensed, MuJoCo is Apache-2.0-licensed, and Minari is MIT-licensed. CIFAR-10 is used as a standard benchmark and is cited to the original dataset source. No new datasets are introduced.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.

Confidential reviewer copy. This manuscript is submitted to the 40th Conference on Neural Information Processing Systems (NeurIPS 2026). Unauthorized sharing, redistribution, or disclosure is strictly prohibited.

- 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.

    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: No human subjects were involved in any way.

    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: No human subjects involved in any experiment.

Confidential reviewer copy. This manuscript is submitted to the 40th Conference on Neural Information Processing Systems (NeurIPS 2026). Unauthorized sharing, redistribution, or disclosure is strictly prohibited.

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: No LLMs are involved in the methods of this research.

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.

Confidential reviewer copy. This manuscript is submitted to the 40th Conference on Neural Information Processing Systems (NeurIPS 2026). Unauthorized sharing, redistribution, or disclosure is strictly prohibited.