WOODBURY INFLUENCE: EFFECTIVE EVALUATION OF UNSEEN SAMPLE FAMILIARITY BEFORE TRAINING

Anonymous authors
Paper under double-blind review

ABSTRACT

Influence Functions (IFs) provide a powerful framework for quantifying the impact of individual training samples on the trained model using gradient and Hessian information. Most IFs primarily focus on the effect of removing samples rather than adding novel ones, but this sample-incremental regime arises in many real-world settings that must determine how familiar a new sample is to the current optimal model. Extending IFs to this regime presents two fundamental challenges: (1) the prohibitive computational cost of estimating influence based on Hessian, and (2) the absence of validated labels in IF frameworks for unseen data. To overcome these challenges, we propose Woodbury Influence (WIF) as a measure of an unseen sample's *familiarity* to the entire model parameters before training on it, leveraging pseudo-labeling and the Woodbury matrix identity. Empirical analyses reveal the underlying mechanisms driving the effective separation of unfamiliar samples from familiar ones, accurate approximation of the true optimal model, and competitive computational cost. Furthermore, we propose practical methods for applying WIFs to representative tasks in this regime, including Out-of-Distribution (OOD) detection, Open-Set Recognition (OSR), and Continual Learning (CL): offline class-order selection. WIF achieves superior performance across all three benchmarks, establishing a robust, influence-based framework for these settings.

1 INTRODUCTION

Influence functions (IFs) have been widely used for evaluating the impact of specific samples on current models after training Koh & Liang (2017); Kwon et al. (2024). Most IF research has focused almost exclusively on this *sample-removal* regime: analyzing how dropping a training sample affects predictions, with applications in data attribution, data valuation, and mislabeled-example detection Pruthi et al. (2020); Koh & Liang (2017); Singh & Alistarh (2020). In contrast, the *sample-incremental* regime has received comparatively little attention. To our knowledge, no prior work has systematically validated against retraining-based ground truth whether IF approximations remain faithful when the query sample lies outside the training distribution, nor provided closed-form implementations tailored to this regime.

Despite this lack of attention, the sample-incremental regime represents a highly practical environment across a variety of applications. For example, a medical diagnostic model routinely faces rare pathologies Roy et al. (2022), an autonomous driving system encounters unseen road conditions Filos et al. (2020), and production classifiers confront entirely new classes Hendrycks et al. (2022). This regime is also central to related fields under various names: Out-of-Distribution (OOD) detection Yang et al. (2024) for identifying if a sample lies outside the training distribution, Open-Set Recognition (OSR) Vaze et al. (2022) for jointly classifying known classes and rejecting unknown ones, and Continual Learning (CL) Wang et al. (2024) for safely incorporating them. Ultimately, all three problems require answering a fundamental question: *Which data are familiar to the current model, and which are not?*

To answer this question, IFs Koh & Liang (2017); Cook & Weisberg (1982) offer strong potential. Because they use both gradient and inverse-Hessian to measure the perturbation that a sample would impose on the optimal model parameters, they can evaluate true model-level impact rather than rely on mere representation-level similarity.

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

Motivated by this potential, this paper extends IFs to the sample-incremental regime, resolving two fundamental difficulties: (i) *Prohibitive computational cost*: computing the inverse-Hessian-vector product (iHvp) relies on iterative solvers such as LiSSA Agarwal et al. (2017) and Conjugate Gradients (CG) Martens (2010) that scale poorly to environments requiring influence estimation over large numbers of samples. (ii) *Absence of validated labels for unseen data*: evaluating the loss gradient at a query sample requires a label, which is inherently unavailable for unseen, inference-stage samples. While the standard workaround in traditional IF frameworks is pseudo-labeling Liu et al. (2021); Pruthi et al. (2020), its behavior in the sample-incremental regime remains poorly understood.

Our contributions are summarized as follows.

1. We propose **WIF (Woodbury Influence)** which reframes influence as familiarity of an unseen sample on the entire parameters and utilizes confidence-based pseudo-labeling with the Woodbury matrix identity to enable efficient, label-free influence estimation in the sample-incremental regime.
2. We empirically validate that WIF successfully generalizes to the sample-incremental regime, faithfully tracking the exact influence.
3. We find empirical evidence of two mechanisms of WIF's effective separation of familiar and unfamiliar unseen samples: the robustness of the pseudo-labeling score distribution against incorrect predictions, and the smoothing of the score distribution via the Hessian-based metric.
4. We present task-specific instantiations of utilizing WIF's effective separation for sample-incremental regimes (OSR, OOD detection, CL : offline class-order selection). Furthermore, experiments across all three benchmarks demonstrate that WIF consistently achieves superior performance, validating its practical effectiveness.

2 Related Work

Influence Functions and Inverse-Hessian Approximation A central computational bottleneck is the inverse-Hessian-vector product (iHVP), for which two representative strategies exist. CG Martens (2010) and LiSSA Agarwal et al. (2017) estimate the inverse iteratively via a truncated Neumann series, incurring substantial overhead from repeated Hessian-vector products. DataInf Kwon et al. (2024) and LoRIF Li et al. (2026) bypass iteration with a closed-form expression by leveraging sherman-morrison identity. however, its approximation error grows with the number of parameters. WoodFisher Singh & Alistarh (2020) similarly leverages the Woodbury identity but for network compression rather than influence estimation.

Out-of-Distribution Detection and Open-Set Recognition A large body of post-hoc OOD detection methods designs gradient-free score functions from network outputs or activations Hendrycks & Gimpel (2017); Liang et al. (2018); Liu et al. (2020); Xu et al. (2024a); Djurisic et al. (2023), and these serve as our primary baselines. A complementary line of work shows that first-order gradient information is itself discriminative: the gradient norm systematically differs between in-distribution and unknown samples, providing a simple yet effective score function Huang et al. (2021); Dauncey et al. (2024). Gradient magnitude alone, however, reflects only how a sample would initially perturb the parameters, while how that perturbation propagates through the *curvature* of the loss landscape, which ultimately determines the effect on model predictions, is encoded in second-order (Hessian) information that gradient-only methods necessarily discard Ye et al. (2021). Influence functions naturally integrate both signals, motivating their use as a unified score for separating familiar from unseen samples.

Continual Learning : offline class-order selection Recent work has shown that the order in which classes or tasks are presented substantially affects catastrophic forgetting, and can be exploited as a curriculum strategy Bell & Lawrence (2022). However, existing approaches determine ordering at the task level and rely on heuristic criteria to identify destructive classes, leaving open the question of how to *quantitatively* distinguish benign from destructive classes at the sample and class level Xu et al. (2024b).

3 PRELIMINARIES

3.1 INFLUENCE FUNCTIONS

Two Forms of Influence Functions Let $\mathcal{X}$ and $\mathcal{Y}$ denote the input and label spaces, and let $\mathcal{D} = \{z_i\}_{i=1}^n$ with $z_i = (x_i, y_i)$ be the training set. For a model $f_\theta : \mathcal{X} \to \mathcal{Y}$ parametrized by $\theta \in \Theta \subseteq \mathbb{R}^d$, the empirical risk minimizer is $\hat{\theta} = \arg\min_\theta \frac{1}{n} \sum_{i=1}^n \ell_i$, where we write $\ell_i = \ell(y_i, f_\theta(x_i))$ for brevity. We further define the per-sample gradient vector $g_i = \nabla \ell_i \in \mathbb{R}^d$.

The influence function framework Cook & Weisberg (1982); Koh & Liang (2017) quantifies the effect of perturbing the contribution of a training sample $z_k \in \mathcal{D}$ on $\hat{\theta}$. Formally, for $\varepsilon \in \mathbb{R}$, we consider the perturbed risk minimizer $\hat{\theta}_{\varepsilon,z_k} = \arg\min_{\theta\in\Theta} \frac{1}{n} \sum_{i=1}^n \ell_i - \varepsilon\, \ell_k$, in which the contribution of z_k is up-weighted by ε. When ℓ is twice differentiable and strongly convex in θ, $\hat{\theta}_{\varepsilon,z_k}$ is well-defined in a neighborhood of $\varepsilon = 0$, and the influence of z_k on $\hat{\theta}$ is defined as the derivative

$$\mathcal{I}_{\hat{\theta}}(z_k) = \left. \frac{d\hat{\theta}_{\varepsilon,z_k}}{d\varepsilon} \right|_{\varepsilon=0} = -H(\hat{\theta})^{-1} \nabla_\theta \ell_k \tag{1}$$

where $H(\hat{\theta}) = \nabla_\theta^2 \big(\frac{1}{n} \sum_{i=1}^n \ell_i \big)$ is the Hessian of the empirical risk. The form of Eq. 1 is influence function of z_k on parameters $\hat{\theta}$ Koh & Liang (2017).

This influence function $\mathcal{I}_{\hat{\theta}}(z_k)$ on parameters captures how z_k moves the entire parameter vector, while a related form measures the resulting effect on a prediction loss evaluated at a validation set $\mathcal{D}_{\text{val}} = \{z_{i,\text{val}}\}_{i=1}^m$:

$$\mathcal{I}_{\hat{\theta}}(z_k, \mathcal{D}_{\text{val}}) = \nabla_\theta \ell_{val}^\top\, H(\hat{\theta})^{-1}\, \nabla_\theta \ell_k, \tag{2}$$

where $\nabla_\theta \ell_{val} = \frac{1}{m} \sum_{i=1}^m \nabla \ell_i$.

Hessian Surrogates and Efficient Inversion Direct computation of $H(\hat{\theta})^{-1}$ is generally infeasible for deep models. When ℓ is a negative log-likelihood, however, Bartlett's second identity Bartlett (1953) gives $\mathbb{E}[\nabla_\theta^2 \ell] = \mathbb{E}[\nabla_\theta \ell\, \nabla_\theta \ell^\top]$, so that the Hessian can be replaced by the Gauss–Newton / Fisher surrogate $\frac{1}{n} \sum_{i=1}^n g_i g_i^\top$ where $g_i = \nabla_\theta \ell_i$. Applying this surrogate to a deep network requires two standard remedies: *damping* Martens (2010), which adds λI to ensure positive definiteness, and a *layer-wise block-diagonal* approximation Grosse et al. (2023), which avoids the prohibitive $d \times d$ memory cost when d reaches millions or billions. Combining both yields

$$-H(\hat{\theta})^{-1} \approx -\sum_{l=1}^{L} \left(\frac{1}{n} \sum_{i=1}^{n} g_{l,i} g_{l,i}^\top + \lambda_l I_{d_l} \right)^{-1} \tag{3}$$

where $\lambda_l > 0$ is the damping coefficient for layer l and d_l is the number of parameters in that layer. DataInf Kwon et al. (2024) sidesteps the $d_l \times d_l$ inversion by swapping inversion and averaging, yielding closed-form rank-one updates:

$$-H(\hat{\theta})^{-1} \approx -\sum_{l=1}^{L} \frac{1}{n} \sum_{i=1}^{n} \left(g_{l,i} g_{l,i}^\top + \lambda_l I_{d_l} \right)^{-1}. \tag{4}$$

This swap is not an identity, and its approximation error grows with d_l, degrading accuracy on large architectures and motivating our Woodbury-based approach in Table. 1.

3.2 WOODBURY MATRIX IDENTITY

The Woodbury matrix identity Woodbury (1950) provides a closed-form expression for the inverse of a matrix. Let $A \in \mathbb{R}^{d\times d}$ be an arbitrary invertible matrix, and let $U \in \mathbb{R}^{d\times k}$, $C \in \mathbb{R}^{k\times k}$, $V \in \mathbb{R}^{k\times d}$ be arbitrary matrices of conforming dimensions, with C and $C^{-1} + VA^{-1}U$ invertible. The identity states

$$(A + UCV)^{-1} = A^{-1} - A^{-1}U \left(C^{-1} + VA^{-1}U \right)^{-1} VA^{-1}. \tag{5}$$

This allows the influence of each incoming sample to be computed in closed form, bypassing the iterative re-inversion that would otherwise be required as samples accumulate.

4 METHOD

4.1 WIF : WOODBURY INFLUENCE

Influence on Entire Parameters as a Familiarity Score To utilize the inspired benefit of IFs for evaluating how familiar of sample $z_\star$ to a model $f_{\hat{\theta}}$, we propose IF as a familiarity score. We define familiarity as the magnitude of influence that $z_\star$ would exert on the optimal parameters if it were incorporated into training:

$$\text{Familiarity}(z) \;:=\; \left\|\mathcal{I}_{\hat{\theta}}(z)\right\| \;=\; \left\|H(\hat{\theta})^{-1}\,\nabla_\theta \ell(z,\hat{\theta})\right\|. \tag{6}$$

A sample whose addition would substantially shift $\hat{\theta}$ is, by this definition, unfamiliar to the model; one that would barely move $\hat{\theta}$ is familiar. We adopt the parameter-based form (Eq. 1) over its loss-based counterpart (Eq. 2) because the latter projects the d-dimensional influence vector onto a single reference direction $\nabla_\theta \ell_{\text{val}}$, tying the score to an arbitrary z_{val}. This projection can collapse to a near-zero scalar even when $z_\star$ has a large footprint, masking genuinely unfamiliar samples. The parameter-based form (Eq. 1), in contrast, is an intrinsic property of $z_\star$ that avoids this cancellation by taking the magnitude directly, consistent with the finding of Lee & AlRegib (2020) that parameter-based gradient quantities separate in- and out-of-distribution samples more reliably than loss-based ones. The performance differences between the two approaches are presented in Table 2 and Table 5.

Pseudo labeling In sample-incremental regime, a fundamental limitation of influence functions $-H_{\hat{\theta}}^{-1}\,\nabla_\theta \ell((x_i,y_i),\hat{\theta})$ is that their formulation requires ground-truth labels y_i, making computation infeasible in settings where true labels are unavailable: $z_\star = (x_\star, \textit{unknown})$ without $y_\star$. To address this, pseudo-labeling approaches have previously been presented Pruthi et al. (2020); Liu et al. (2021). To enable the use of influence functions even in such unlabeled scenarios, we apply a pseudo-labeling strategy that slightly modifies the computation of gradient as needed.

Given the confidence vector $f_c(x_\star) \in [0,1]^K$ over K classes, we obtain the hard prediction $y_\star(x_\star) \in \{0,1\}^K$ by applying a one-hot transformation to the $\arg\max$ operation:

$$y_{\star,pseudo} = \text{onehot}\left(\underset{k\in\{1,\ldots,K\}}{\arg\max}\,\left[f_c(x_\star)\right]_k\right), \tag{7}$$

where $\left[f_c(x_\star)\right]_k$ denotes the k-th component of the confidence vector, corresponding to the predicted probability of class k. This approach enables influence function in more broad settings where true labels are unknown. Section 5.2 provides empirical evidences of valid behavior of this labeling as an approximation of the change of optimal model parameters for unseen sample.

Woodbury Matrix Identity for the Inverse Hessian Computing the inverse Hessian via iHVP methods such as CG or LiSSA incurs substantial computational cost, making them impractical when influence values must be computed for many query samples. We instead exploit the low-rank structure of the empirical Fisher surrogate via the Woodbury matrix identity Singh & Alistarh (2020).

For each layer l, we collect the per-sample gradients $\{g_{l,i}\}_{i=1}^n$ into the (normalized) matrix $G_l := \frac{1}{\sqrt{n}}[g_{l,1}, g_{l,2}, \ldots, g_{l,n}] \in \mathbb{R}^{d_l \times n}$ so that the empirical-Fisher surrogate of Eq. 3 admits the compact form $\frac{1}{n}\sum_{i=1}^n g_{l,i}\, g_{l,i}^\top = G_l G_l^\top$. The layer-$l$ damped surrogate thus becomes $G_l G_l^\top + \lambda I_{d_l}$.

Then, applying the Woodbury matrix identity shown in Eq. 5 by letting $A = \lambda I_{d_l}$, $U = G_l$, $C = I_n$, $V = G_l^\top$, we obtain the following closed-form inverse:

$$\left(G_l G_l^\top + \lambda I_{d_l}\right)^{-1} = \tfrac{1}{\lambda} I_{d_l} - \tfrac{1}{\lambda^2}\, G_l\, \mathcal{M}\, G_l^\top, \quad \mathcal{M} := \left(I_n + \tfrac{1}{\lambda} G_l^\top G_l\right)^{-1}. \tag{8}$$

The decisive saving is dimensional: $\mathcal{M}$ is the inverse of an $n \times n$ matrix, whereas the left-hand side is a $d_l \times d_l$ inverse. When $d_l \gg n$, this significantly reduces the computation cost of an inversion $\mathcal{O}(d_l^2)$ into an $\mathcal{O}(n^2)$ one.

Then, we build WIF by combining Eq. 3 and Eq. 8 yields the WIF estimate

$$\mathcal{I}_{\text{WIF}}(z_\star) = -\sum_{l=1}^{L}\left[\tfrac{1}{\lambda}\, g_\star^{(l)} - \tfrac{1}{\lambda^2}\, G_l\, \mathcal{M}\, G_l^\top\, g_\star^{(l)}\right], \tag{9}$$

where $g_\star = \nabla_\theta \ell(z_\star, \hat{\theta})$ with $z_\star \notin \mathcal{D}$, and $g_\star^{(l)}$ denotes the layer-l block of $g_\star$.

4.2 WIF'S APPLICATION

OOD Detection and OSR A wide range of methods score samples for OOD detection using network outputs, activations, or gradients Xu et al. (2024a); Liu et al. (2020); Djurisic et al. (2023); Hendrycks & Gimpel (2017); Huang et al. (2021); Liang et al. (2018). WIF instantiates such a score through the magnitude of the influence vector, distinguishing ID from OOD samples by comparing $\|\mathcal{I}_{\text{WIF}}(z_\star)\|$ against a threshold. Following GradNorm Huang et al. (2021), which finds last-layer gradient information most discriminative for OOD detection, we set $L = 1$ in Eq. 9 and use only the final-layer block of $g_\star$ and G_l, which also reduces the per-query cost. The resulting WIF score is

$$S_{\textbf{WIF}}(z_\star) = \left\| H(\hat{\theta})^{-1} g_\star \right\| = \left\| \tfrac{1}{\lambda} g_\star - \tfrac{1}{\lambda^2} G\, \mathcal{M}\, G^\top g_\star \right\|, \tag{10}$$

where G and $\mathcal{M}$ are the last-layer instances of the quantities defined in Eq. 8, and $g_\star$ denotes the last-layer block of the query gradient. We verify empirically in Sec. 5.4 that combining gradient direction with local curvature outperforms purely gradient-based scores.

Class incremental Continual Learning. Task ordering has a significant impact on forgetting, and can be exploited as a curriculum strategy Bell & Lawrence (2022). However, the criteria for determining which classes are **destructive** remain unclear Xu et al. (2024b), necessitating a quantitative metric that can numerically distinguish between benign and destructive classes. Based on the above studies, we adopted IF as a quantitative metric for distinguishing between benign and destructive classes. Let $\mathcal{D}_k = \{x_i : y_i = k\}$ denote the set of samples belonging to class $k \in \{1, \ldots, K\}$. For each sample, we compute the WIF score, yielding a per-class score distribution $\mathcal{S}_k = \{\text{WIF}(x_i) \,|\, x_i \in \mathcal{D}_k\}$ .. To mitigate the effect of outliers, we adopt a 10% trimmed mean. Let $q_\alpha(\mathcal{S}_k)$ denote the α-quantile of $\mathcal{S}_k$. The robust class-level score is defined as

$$\bar{s}_k = \frac{1}{|\mathcal{S}_k^{\text{trim}}|} \sum_{s \in \mathcal{S}_k^{\text{trim}}} s, \quad \mathcal{S}_k^{\text{trim}} = \{s \in \mathcal{S}_k \,|\, q_{0.1}(\mathcal{S}_k) \le s \le q_{0.9}(\mathcal{S}_k)\}. \tag{11}$$

Classes are then ranked in ascending order of $\bar{s}_k$, producing an ordering from the most benign to the most destructive:

$$\pi = \text{argsort}_{k \in \{1,\ldots,K\}} (\bar{s}_k). \tag{12}$$

5 EXPERIMENTS

We evaluate WIF through 5 sets of experiments: (i) approximation quality validation in the sample-incremental regime (Sec. 5.1), (ii) ablation of Hessian and pseudo-labeling components (Sec. 5.2), (iii) downstream performance on OOD detection and OSR (Sec. 5.4–5.5), and (iv) application to Continual learning (Sec. 5.6).

5.1 IS APPROXIMATION OF WIF ACCURATE?

Experimental Settings. We validate approximation quality of WIF in the sample-incremental regime through two complementary experiments. For primary validation Fig. 1, we train a CNN on MNIST dataset using only five of the ten classes; the remaining five classes serve as out-of-training-set samples, enabling us to measure IF behavior on genuinely unknown data rather than held-out ID samples. We uniformly sample 20 instances from each of the five unknown classes (100 samples total) and, for each, compute (i) the exact influence $-H_{\hat{\theta}}^{-1} \nabla_\theta \ell(z_\star)$ (Eq. 3) obtained by exactly retraining the model on $\mathcal{D} \cup \{z_\star\}$, and (ii) the WIF estimate $-H_{\text{Wood}}^{-1} \nabla_\theta \ell(z_\star)$ (Eq. 9) calculated without retraining. For the verifying the

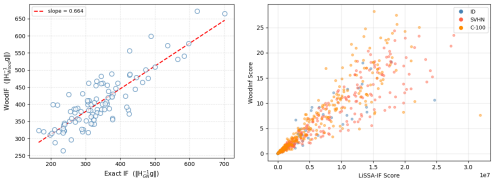


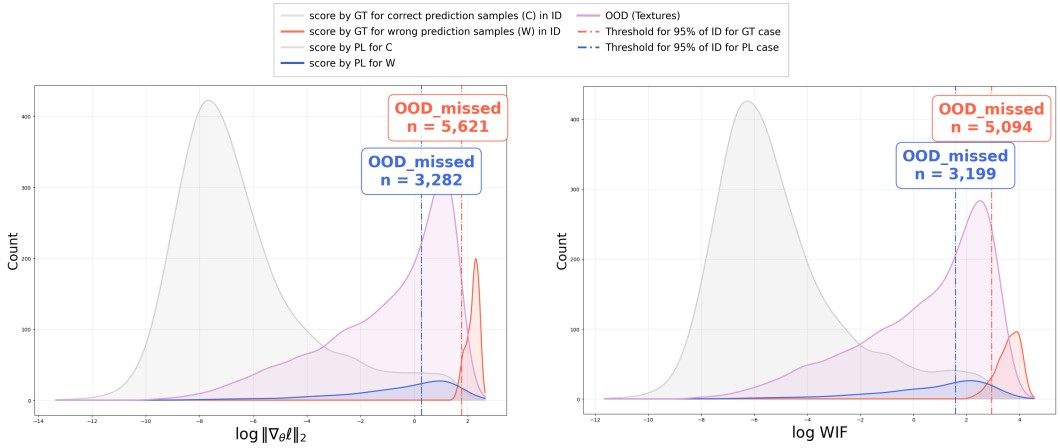
Figure 1: **Approximation Quality Validation of WIF in the sample-incremental regime. Left:** WIF estimates $\|H_{\text{Wood}}^{-1}\, g\|$ vs. exact IF values $\|H_{\hat{\theta}}^{-1}\, g\|$ on 100 out-of-training-set samples (CNN on MNIST, five unknown classes). **Right:** WIF vs. LiSSA agreement on ResNet-18 / CIFAR-10, evaluated on 300 query samples.

scalability of the validation, we train ResNet-18
on CIFAR-10 and compare WIF against LiSSA-based IF estimates on a mixture of 100 ID samples, 100 near-OOD samples (CIFAR-100), and 100 far-OOD samples (SVHN), totaling 300 query points. A log transformation was applied to linearize the nonlinear relationship in the correlation analysis.Box & Cox (1964).

Incrementality : Sample Removal vs Sample Addition Although Eq 1 is classically derived under the assumption that $z_i \in \mathcal{D}$, we test whether WIF remains faithful when $z_\star \notin \mathcal{D}$. As shown in Fig. 1 (left), WIF estimates track the exact values closely across the full range of influence magnitudes, yielding **Pearson** $r = 0.85$ and **Spearman** $\rho = 0.78$ on samples never seen during training, providing empirical evidence that the IF framework remains meaningful in the sample-incremental regime. The right panel extends this validation to ResNet-18 on CIFAR-10 against LiSSA-based IF, achieving Pearson $r = 0.92$ and Spearman $\rho = 0.98$ and confirming that WIF's approximation accuracy is comparable to LiSSA's at deeper architectures. (Theoretical justification is provided in Appendix A.)

5.2 Why WIF enhances the separation of ID from OOD samples within unseen data?

Experimental Settings. To disentangle the contributions of the two core components of WIF – the Hessian-based curvature term and the pseudo-labeling scheme – we analyze score distributions on CIFAR-10 (N=10,000 ID test samples) against Textures dataset as OOD (N=5,640). We further split the ID samples by prediction correctness: *correct* samples (N=9,473), where the model's prediction matches the true label, and *wrong* samples (N=527), where it does not. We compare four variants on a log scale: (i) gradient norm with ground-truth (GT) labels, (ii) gradient norm with pseudo labels, (iii) WIF with ground-truth (GT) labels, and (iv) WIF with pseudo labels. For each variant we set the *ID 95% threshold* – the score below which 95% of ID samples fall – and report *OOD-missed*, the number of OOD samples scoring below that threshold (lower is better).



(a) Sample count by gradient norm ($\log \|\nabla_\theta \ell\|$). Pseudo-labeling lowers *OOD-missed* samples from 5,621 to 3,282 for incorrectly predicted ID samples.

(b) Sample count by WIF ($\log \| - H^{-1} \nabla_\theta \ell\|$). Hessian further reduces *OOD-missed* for ID samples compared to gradient norm in (a).

Figure 2: Impact of pseudo-labeling and Hessian to ID sample distribution. Both Pseudo-labeling and the Hessian reduce *OOD-missed* especially for wrong prediction ID samples. Score distributions on CIFAR-10 (ID) vs. Textures (OOD), split by ID prediction correctness (correct: gray, wrong: red/blue) and OOD (purple). Vertical lines mark the ID 95% threshold for the true-label (red dashed line) and pseudo-label (blue dashed line) variants. (PL: pseudo-label, GT: ground-truth)

Effect of pseudo-labeling. Within each panel, pseudo-labeled scores on wrong samples (blue) are systematically shifted left relative to true-labeled scores (red), dragging the ID 95% threshold toward the OOD distribution and reducing *OOD-missed*. With gradient-norm scoring, *OOD-missed* drops from 5,621 to 3,282 (-41.6%); with WIF, it drops from 5,094 to 3,199 (-37.2%). The reason is

alignment: at inference, OOD samples have no ground-truth label, so scoring ID samples with their *predicted* labels produces a consistent regime that matches the OOD evaluation pipeline, yielding cleaner ID–OOD separation.

Effect of the Hessian. Misclassified inputs produce steeper loss gradients, inflating the wrong-sample distribution and pushing the 95% threshold rightward so that OOD samples slip below it. The Hessian term in WIF rescales scores by local curvature, dampening this inflation and pulling wrong samples toward the correct-sample distribution. This reduces ***OOD-missed*** from $5{,}621$ to $5{,}094$ under true labels and from $3{,}282$ to $3{,}199$ under pseudo labels; combined with pseudo-labeling, the two mechanisms yield a 43.1% reduction over gradient norm with true labels, translating into the downstream gains in Sec. 5.4.

5.3 Complexity and Performance Trade-Off

DataInf Kwon et al. (2024) differs from WIF on two axes that translate into measurable gaps in both accuracy and speed.

Accuracy gap. DataInf's swap of inversion and averaging (Eq. 4) is not an identity and incurs error that grows with layer width d_l. WIF instead applies the Woodbury identity directly to $G_l G_l^\top + \lambda I_{d_l}$ without any additional approximation, preserving the empirical-Fisher structure. This yields downstream gains on OOD (Table 2 and Table 5).

Computational gap. As shown in Table 1, while WIF is somewhat less favorable than DataInf in terms of memory complexity (space complexity), it offers a slight advantage in computational complexity (time complexity). This speed advantage arises because DataInf estimates the loss-based influence $\nabla_\theta \ell_{\text{val}}^\top H^{-1} g_\star$ of Eq. 2, requiring an extra gradient and inner product per query, whereas WIF directly outputs $H^{-1} g_\star$ (parameter-based IF). As a result, WIF is slightly faster in practice. Even in the regime where $n \gg d$, WIF leverages the push-through identity trick to construct $\mathcal{M}$ using the smaller of n and d, thereby achieving a memory advantage (see Appendix B.2).

Table 1: **Time and Space Complexity of influence estimation methods.** We denote the number of reference samples by n, and parameter dimension by d. Total time is computed as n * per-query cost. Exact IF is intractable at this scale and is therefore excluded from the timing comparison.

Method	Hessian inversion	Time complexity	Memory complexity	Total time/sample (s)
Exact IF	Matrix inversion	$\mathcal{O}(nd^2 + d^3)$	$\mathcal{O}(d^2)$	–
LiSSA w/ pseudo	Iterative	$\mathcal{O}(nd^2)$	$\mathcal{O}(d^2)$	10
DataInf w/ pseudo	Closed-form (Additional approximation)	$\mathcal{O}(nd)$	$\mathcal{O}(d)$	0.020
WIF (ours)	Closed-form (Woodbury)	$\mathcal{O}(nd)$	$\mathcal{O}(\min(n,d)^2)$	**0.018**

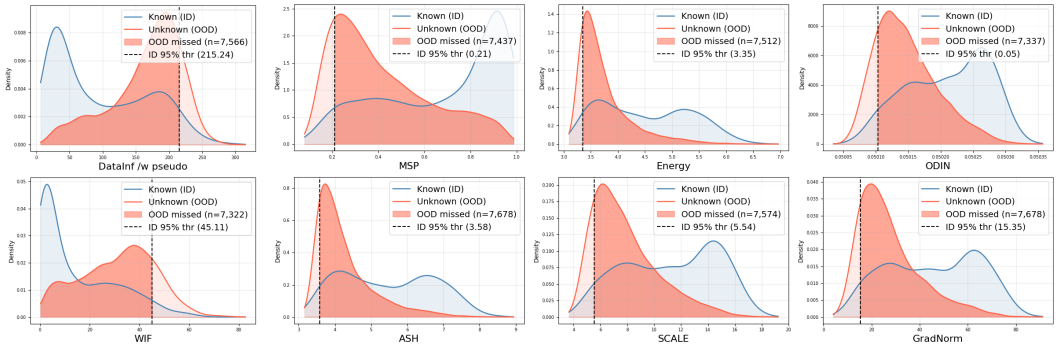
5.4 Out-of-Distribution Detection

Experimental Settings. We evaluate WIF on the OOD detection task using ResNet-18 trained on Tiny ImageNet, comparing it against representative post-hoc baselines spanning two method families: *gradient-free* (MSP Hendrycks & Gimpel (2017), Energy Liu et al. (2020), ODIN Liang et al. (2018), ASH Djurisic et al. (2023), SCALE Xu et al. (2024a)) and *gradient-based* (GradNorm Huang et al. (2021)). WIF extends beyond these families by additionally incorporating inverse Hessian information. Following the standard OOD detection protocol, we set the decision threshold at the 95-th percentile of ID scores, which is a widely used operating point that fixes the true-positive rate of ID acceptance at 95%. We report AUROC and FPR95 across four OOD benchmarks: SVHN and Textures (far-OOD), CIFAR-10 and CIFAR-100 (near-OOD). Additional experiments (ViT backbone trained on Tiny ImageNet) are provided in the Appendix B.1.

Detection performance. Table 2 summarizes the OOD detection results.WIF achieves the best FPR95 across all four benchmarks. We discuss why WIF particularly excels at FPR95 in Sec. 5.5.

Table 2: **OOD detection results on ResNet-18 trained on Tiny ImageNet (AUROC ↑ / FPR95 ↓).** Results are obtained by fine-tuning only the last layer of the pretrained model, averaged over 5 seeds. Best results in **bold**.

Method space	Method	SVHN (Far-OOD)		Textures (Far-OOD)		CIFAR-10 (Near-OOD)		CIFAR-100 (Near-OOD)	
		AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓
No Gradient	MSP	$0.885_{\pm 0.003}$	$0.581_{\pm 0.020}$	$0.717_{\pm 0.013}$	$0.892_{\pm 0.041}$	$0.784_{\pm 0.008}$	$0.817_{\pm 0.020}$	$0.781_{\pm 0.007}$	$0.817_{\pm 0.007}$
	Energy	$\mathbf{0.923}_{\pm 0.004}$	$0.431_{\pm 0.021}$	$0.722_{\pm 0.012}$	$0.875_{\pm 0.037}$	$\mathbf{0.792}_{\pm 0.007}$	$0.755_{\pm 0.018}$	$0.803_{\pm 0.002}$	$0.717_{\pm 0.008}$
	ODIN	$0.918_{\pm 0.007}$	$0.484_{\pm 0.028}$	$0.723_{\pm 0.021}$	$0.891_{\pm 0.038}$	$0.784_{\pm 0.007}$	$0.804_{\pm 0.025}$	$0.791_{\pm 0.005}$	$0.783_{\pm 0.013}$
	ASH	$0.875_{\pm 0.007}$	$0.780_{\pm 0.027}$	$0.701_{\pm 0.011}$	$0.901_{\pm 0.041}$	$0.754_{\pm 0.009}$	$0.828_{\pm 0.024}$	$0.764_{\pm 0.002}$	$0.822_{\pm 0.005}$
	SCALE	$0.881_{\pm 0.008}$	$0.643_{\pm 0.028}$	$0.673_{\pm 0.015}$	$0.930_{\pm 0.058}$	$0.738_{\pm 0.004}$	$0.839_{\pm 0.012}$	$0.740_{\pm 0.003}$	$0.841_{\pm 0.017}$
Gradient	GradNorm	$0.731_{\pm 0.005}$	$0.912_{\pm 0.017}$	$0.651_{\pm 0.031}$	$0.911_{\pm 0.068}$	$0.699_{\pm 0.005}$	$0.886_{\pm 0.010}$	$0.671_{\pm 0.003}$	$0.867_{\pm 0.009}$
Gradient + Hessian	DataInf /w pseudo	$0.887_{\pm 0.003}$	$0.398_{\pm 0.015}$	$0.718_{\pm 0.016}$	$0.697_{\pm 0.056}$	$0.782_{\pm 0.010}$	$0.596_{\pm 0.023}$	$0.794_{\pm 0.001}$	$0.652_{\pm 0.003}$
	WIF (ours)	$0.888_{\pm 0.002}$	$\mathbf{0.391}_{\pm 0.015}$	$\mathbf{0.733}_{\pm 0.016}$	$\mathbf{0.685}_{\pm 0.057}$	$0.788_{\pm 0.010}$	$\mathbf{0.591}_{\pm 0.023}$	$\mathbf{0.805}_{\pm 0.001}$	$\mathbf{0.631}_{\pm 0.003}$



(a) DataInf and WIF (b) Baselines

Figure 3: **Sample Count by Familiarity Score for Comparison of ID (blue) and OOD (red) sets on Tiny ImageNet.** The dashed vertical line marks the ID 95-th-percentile threshold; the shaded red region indicates OOD samples on the OOD side of this threshold (lower is better). Note that the scoring convention is reversed between the two panels: for WIF and DataInf (a) larger scores indicate OOD, while for the baselines (b) smaller scores indicate OOD. Hence the threshold-determining tail is the right tail of the ID distribution in (a) and the left tail in (b).

5.5 Open-Set Recognition

Experimental Settings. We evaluate WIF on the OSR task using Tiny ImageNet with ResNet-18. we train the model on 20 known classes and treat the remaining 180 classes as *unknown*. At test time, the model must both (i) classify known samples correctly and (ii) assign low confidence to unknown samples. We report AUROC and FPR95 for the known-vs-unknown separation task, using the same set of baselines as in Sec. 5.4.

OSR performance. Table 3 reports OSR results. WIF achieves the highest AUROC (0.7838) and the lowest FPR95 (0.5700) among all compared methods. The gap is particularly pronounced on FPR95, where WIF – the method that incorporates Hessian information – dominates all gradient-only and no gradient-baselines by a wide margin (0.57 vs. 0.82+ for others). This confirms that curvature information is especially valuable in the OSR setting, where known and unknown classes share low-level visual statistics but differ in how they interact with the model's learned decision boundary – a distinction that gradient magnitude alone cannot capture but that inverse-Hessian-weighted influence reliably detects.

Table 3: **Open-Set Recognition on Tiny ImageNet.** AUROC (↑) and FPR95 (↓) for known-vs-unknown separation. Model trained on 20 known classes; 180 held-out classes serve as unknown.Results are averaged over 5 runs with different known/unknown class splits.

Method space	Method	Tiny ImageNet	
		AUROC ↑	FPR95 ↓
No Gradient	MSP	$0.776_{\pm 0.011}$	$0.826_{\pm 0.081}$
	Energy	$0.779_{\pm 0.020}$	$0.834_{\pm 0.071}$
	ODIN	$0.782_{\pm 0.013}$	$0.815_{\pm 0.070}$
	ASH	$0.778_{\pm 0.010}$	$0.853_{\pm 0.078}$
	SCALE	$0.779_{\pm 0.012}$	$0.841_{\pm 0.061}$
Gradient	GradNorm	$0.773_{\pm 0.016}$	$0.853_{\pm 0.082}$
Gradient + Hessian	WIF (ours)	$\mathbf{0.784}_{\pm 0.012}$	$\mathbf{0.570}_{\pm 0.030}$

Why WIF excels at FPR95. Since FPR95 is set by the 95-th percentile of ID scores, the decisive factor is how gradually the ID distribution decays at the threshold-determining tail: a steep drop pushes the threshold into the OOD-dense region and inflates FPR95, whereas a smooth decay places it farther from the OOD mass. Fig. 3 shows that **WIF is the only method whose ID distribution decays smoothly through the cutoff**, while every competing method exhibits a steep falloff or bimodal shape in the relevant tail. This complements the ablation in Sec. 5.2: Hessian-induced rescaling suppresses the tail irregularity that gradient-only and no-gradient baselines cannot.

5.6 Continual Learning : offline class-order selection

We apply WIF to class-incremental Continual Learning (CL), where the model incrementally acquires new classes without forgetting previously learned ones. Task ordering is known to affect catastrophic forgetting Bell & Lawrence (2022), but existing criteria for identifying destructive classes operate at the task level and remain heuristic Xu et al. (2024b). WIF supplies a quantitative criterion at the *class level*: which classes to learn first.

Experimental Settings. We evaluate on Tiny ImageNet with ResNet-18. The model is trained on 5 base classes and then incrementally learns 5 more, one class per step. We evaluate two representative CL methods spanning the major families: EWC Kirkpatrick et al. (2016) (regularization-based) and GEM Lopez-Paz & Ranzato (2017) (replay-based). Performance is measured by Average Accuracy (AA, ↑) and Forgetting Measure (FM, ↓) averaged over all encountered classes.

Class-level curriculum via WIF. To determine the incremental class order, we compute the WIF score for every sample in each candidate class and aggregate to a class-level score. To mitigate the influence of outlier samples, we trim the top and bottom 10% of the within-class WIF distribution and use the trimmed mean as the class score (Eq. 11, 12). We then define two ordering strategies: **Ben-first**, where classes with lower trimmed-mean scores are introduced first (benign-first), and **Des-first**, where classes with higher scores come first (destructive-first). The **Random** order follows the default dataset class indexing and serves as a baseline. For comparison, we additionally include a logit-norm-based ordering, restricted to the Ben-first order (high-logit-first), following the standard curriculum-learning practice of introducing confident samples first.

CL performance results are shown in Table 4. In ordering comparison, across both EWC and GEM, benign-first ordering consistently outperforms destructive-first ordering and the default ordering on both accuracy and forgetting. This confirms that the ordering effect is not merely a matter of any structured curriculum but specifically of the *direction* dictated by WIF.

Table 4: **Class-incremental CL on Tiny ImageNet (ResNet-18, 5+5 classes).** Average Accuracy (AA, ↑) and Forgetting Measure (FM, ↓) under four class-ordering strategies. Averaged over 5 seeds.

	EWC		GEM	
Order	AA↑	FM↓	AA↑	FM↓
Random	$0.40_{\pm 0.08}$	$0.60_{\pm 0.10}$	$0.54_{\pm 0.09}$	$0.46_{\pm 0.10}$
Des-first /w WIF	$0.37_{\pm 0.03}$	$0.58_{\pm 0.05}$	$0.62_{\pm 0.03}$	$0.44_{\pm 0.05}$
Ben-first /w Logit	$0.41_{\pm 0.03}$	$0.56_{\pm 0.04}$	$0.72_{\pm 0.03}$	$0.32_{\pm 0.06}$
Ben-first /w WIF	$\mathbf{0.42}_{\pm 0.02}$	$\mathbf{0.54}_{\pm 0.04}$	$\mathbf{0.74}_{\pm 0.02}$	$\mathbf{0.25}_{\pm 0.05}$

In performance comparison on both GEM and EWC, WIF with Ben-first ordering improves AA and FM compared to the default ordering, surpassing logit-based ordering as well. This validates that low-WIF classes induce milder parameter perturbations and form a natural curriculum foundation, whereas destructive classes cause abrupt loss-landscape changes when encountered early.

6 Conclusion

We proposed Woodbury Influence (WIF), a measure of how familiar an unseen sample is to a trained model, computed without any additional training. Combining the Woodbury matrix identity with confidence-based pseudo-labeling, WIF removes the two obstacles that have kept Influence Functions out of this regime: the cost of inverse-Hessian-vector products and the absence of validated labels for unseen data. WIF tracks exact influence values, matches LiSSA's accuracy at a fraction of its cost, and outperforms baselines on OOD detection, open-set recognition, and class-incremental continual learning : offline class-order selection. WIF thus turns the Influence Function from a training-data diagnostic into an efficient familiarity measure for open-world learning.

AI USE STATEMENT

We used generative AI tools to *assist with translation*: portions of the manuscript were drafted in the authors' first language and translated into English with LLM assistance, then revised by the authors. We did not use generative AI tools for the remaining required-disclosure tasks that apply to this work, namely developing the theoretical framework, formulating the mathematical claims underlying the Woodbury-based estimator, writing their proofs, proposing or refining the hypotheses, designing the experiments, and interpreting the results. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Naman Agarwal, Brian Bullins, and Elad Hazan. Second-order stochastic optimization for machine learning in linear time. *Journal of Machine Learning Research*, 18(116):1–40, 2017.

M. S. Bartlett. Approximate confidence intervals. *Biometrika*, 40(1/2):12–19, 1953.

Samuel J. Bell and Neil D. Lawrence. The effect of task ordering in continual learning. *arXiv preprint arXiv:2205.13323*, 2022.

George E. P. Box and David R. Cox. An analysis of transformations. *Journal of the Royal Statistical Society: Series B (Methodological)*, 26(2):211–252, 1964.

R. Dennis Cook and Sanford Weisberg. *Residuals and Influence in Regression*. Chapman and Hall, New York, 1982.

Sam Dauncey, Christopher C. Holmes, Christopher Williams, and Fabian Falck. Approximations to the fisher information metric of deep generative models for out-of-distribution detection. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=EcuwtinFs9.

Andrija Djurisic, Nebojsa Bozanic, Arjun Ashok, and Rosanne Liu. Extremely simple activation shaping for out-of-distribution detection. In *International Conference on Learning Representations (ICLR)*, 2023.

Angelos Filos, Panagiotis Tigas, Rowan McAllister, Nicholas Rhinehart, Sergey Levine, and Yarin Gal. Can autonomous vehicles identify, recover from, and adapt to distribution shifts? In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, volume 119 of *Proceedings of Machine Learning Research*, pp. 3145–3153. PMLR, 2020.

Roger Grosse, Juhan Bae, Cem Anil, Nelson Elhage, Alex Tamkin, Amirhossein Tajdini, Benoit Steiner, Dustin Li, Esin Durmus, Ethan Perez, Evan Hubinger, Kamilė Lukošiūtė, Karina Nguyen, Nicholas Joseph, Sam McCandlish, Jared Kaplan, and Samuel R. Bowman. Studying large language model generalization with influence functions. *arXiv preprint arXiv:2308.03296*, 2023.

Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *International Conference on Learning Representations (ICLR)*, 2017.

Dan Hendrycks, Steven Basart, Mantas Mazeika, Andy Zou, Joseph Kwon, Mohammadreza Mostajabi, Jacob Steinhardt, and Dawn Song. Scaling out-of-distribution detection for real-world settings. In *Proceedings of the 39th International Conference on Machine Learning (ICML)*, volume 162 of *Proceedings of Machine Learning Research*, pp. 8759–8773. PMLR, 2022.

Rui Huang, Andrew Geng, and Yixuan Li. On the importance of gradients for detecting distributional shifts in the wild. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=fmiwLdJCmLS.

James Kirkpatrick, Razvan Pascanu, Neil C. Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. Overcoming catastrophic forgetting in neural networks. *CoRR*, abs/1612.00796, 2016. URL http://arxiv.org/abs/1612.00796.

Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, pp. 1885–1894, 2017.

Yongchan Kwon, Eric Wu, Kevin Wu, and James Zou. DataInf: Efficiently estimating data influence in LoRA-tuned LLMs and diffusion models. In *International Conference on Learning Representations*, 2024.

Jinsol Lee and Ghassan AlRegib. Gradients as a measure of uncertainty in neural networks. In *IEEE International Conference on Image Processing (ICIP)*, pp. 2826–2830. IEEE, 2020.

Shuangqi Li, Hieu Le, Jingyi Xu, and Mathieu Salzmann. Lorif: Low-rank influence functions for scalable training data attribution, 2026. URL https://arxiv.org/abs/2601.21929.

Shiyu Liang, Yixuan Li, and R. Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. In *International Conference on Learning Representations (ICLR)*, 2018.

Weitang Liu, Xiaoyun Wang, John Owens, and Yixuan Li. Energy-based out-of-distribution detection. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Zhuoming Liu, Hao Ding, Huaping Zhong, Weijia Li, Jifeng Dai, and Conghui He. Influence selection for active learning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 9274–9283, 2021.

David Lopez-Paz and Marc'Aurelio Ranzato. Gradient Episodic Memory for Continual Learning. In *Advances in Neural Information Processing Systems 30 (NIPS 2017)*, pp. 1–11, 2017. URL https://papers.nips.cc/paper_files/paper/2017/file/f87522788a2be2d171666752f97ddebb-Paper.pdf.

James Martens. Deep learning via Hessian-free optimization. In *Proceedings of the 27th International Conference on Machine Learning (ICML)*, pp. 735–742, Haifa, Israel, 2010. Omnipress.

Garima Pruthi, Frederick Liu, Satyen Kale, and Mukund Sundararajan. Estimating training data influence by tracing gradient descent. In *Advances in Neural Information Processing Systems*, volume 33, pp. 19920–19930, 2020.

Abhijit Guha Roy, Jie Ren, Shekoofeh Azizi, Aaron Loh, Vivek Natarajan, Basil Mustafa, Nick Pawlowski, Jan Freyberg, Yuan Liu, Zach Beaver, Nam Vo, Peggy Bui, Samantha Winter, Patricia MacWilliams, Greg S. Corrado, Umesh Telang, Yun Liu, Taylan Cemgil, Alan Karthikesalingam, Balaji Lakshminarayanan, and Jim Winkens. Does your dermatology classifier know what it doesn't know? detecting the long-tail of unseen conditions. *Medical Image Analysis*, 75:102274, 2022. doi: 10.1016/j.media.2021.102274.

Sidak Pal Singh and Dan Alistarh. WoodFisher: Efficient second-order approximation for neural network compression. In *Advances in Neural Information Processing Systems*, volume 33, pp. 18098–18109, 2020.

Sagar Vaze, Kai Han, Andrea Vedaldi, and Andrew Zisserman. Open-set recognition: a good closed-set classifier is all you need? In *International Conference on Learning Representations*, 2022.

Liyuan Wang, Xingxing Zhang, Hang Su, and Jun Zhu. A comprehensive survey of continual learning: Theory, method and application. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(8):5362–5383, 2024. doi: 10.1109/TPAMI.2024.3367329.

Max A. Woodbury. Inverting modified matrices. Technical Report Memorandum Report 42, Statistical Research Group, Princeton University, Princeton, NJ, 1950.

Kai Xu, Rongyu Chen, Gianni Franchi, and Angela Yao. Scaling for training time and post-hoc out-of-distribution detection enhancement. In *International Conference on Learning Representations (ICLR)*, 2024a.

Shixiong Xu, Gaofeng Meng, Xing Nie, Bolin Ni, Bin Fan, and Shiming Xiang. Defying imbalanced forgetting in class incremental learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 2024b.

Jingkang Yang, Kaiyang Zhou, Yixuan Li, and Ziwei Liu. Generalized out-of-distribution detection: A survey. *International Journal of Computer Vision*, 132(12):5635–5662, 2024. doi: 10.1007/s11263-024-02117-4.

Haotian Ye, Chuanlong Xie, Yue Liu, and Zhenguo Li. Out-of-distribution generalization analysis via influence function, 2021. URL https://openreview.net/forum?id=KcTBbZ1kM6K.

A Appendix A : Theoretical Proof

A.1 Theoretical Justification for Sample Addition

Let

$$R_n(\theta) = \frac{1}{n}\sum_{i=1}^{n} \ell(z_i, \theta), \qquad \hat{\theta}_n = \arg\min_{\theta} R_n(\theta).$$

For an arbitrary query sample $z_\star$, not necessarily drawn from the training distribution, define the empirical risk after adding $z_\star$ as

$$R_{n,+z_\star}(\theta) = \frac{1}{n+1}\left(\sum_{i=1}^{n} \ell(z_i, \theta) + \ell(z_\star, \theta)\right).$$

Since multiplication of an objective by a positive constant does not change its minimizer,

$$\hat{\theta}_{n,+z_\star} = \arg\min_{\theta}\left[R_n(\theta) + \frac{1}{n}\ell(z_\star, \theta)\right].$$

Theorem 1 (Influence of an arbitrary added sample)**.** *Assume that R_n and $\ell(z_\star, \cdot)$ are twice continuously differentiable in a neighborhood of $\hat{\theta}_n$, and that*

$$\nabla R_n(\hat{\theta}_n) = 0, \qquad H_n := \nabla^2 R_n(\hat{\theta}_n) \succ 0.$$

Then there exists a locally unique differentiable path of local minimizers $\theta_\star(\epsilon)$ of

$$R_n(\theta) + \epsilon\ell(z_\star, \theta)$$

such that $\theta_\star(0) = \hat{\theta}_n$, and

$$\left.\frac{d\theta_\star(\epsilon)}{d\epsilon}\right|_{\epsilon=0} = -H_n^{-1}\nabla\ell(z_\star, \hat{\theta}_n).$$

Consequently,

$$\hat{\theta}_{n,+z_\star} - \hat{\theta}_n = -\frac{1}{n}H_n^{-1}\nabla\ell(z_\star, \hat{\theta}_n) + o(n^{-1}).$$

If the corresponding Hessians are locally Lipschitz continuous, the remainder is $O(n^{-2})$.

Proof. Define

$$F(\theta, \epsilon) = \nabla R_n(\theta) + \epsilon\nabla\ell(z_\star, \theta).$$

At $(\hat{\theta}_n, 0)$,

$$F(\hat{\theta}_n, 0) = 0, \qquad \nabla_\theta F(\hat{\theta}_n, 0) = H_n.$$

Since H_n is invertible, the implicit function theorem guarantees the existence of a locally unique differentiable function $\theta_\star(\epsilon)$ satisfying $F(\theta_\star(\epsilon), \epsilon) = 0$. Differentiating this identity with respect to ϵ gives

$$\nabla_\theta F(\theta_\star(\epsilon), \epsilon)\theta'_\star(\epsilon) + \nabla\ell(z_\star, \theta_\star(\epsilon)) = 0.$$

Evaluating at $\epsilon = 0$ yields

$$H_n\theta'_\star(0) + \nabla\ell(z_\star, \hat{\theta}_n) = 0,$$

and hence

$$\theta'_\star(0) = -H_n^{-1}\nabla\ell(z_\star, \hat{\theta}_n).$$

The result follows by a Taylor expansion at $\epsilon = 0$ and by substituting $\epsilon = 1/n$. Importantly, the argument does not require $z_\star$ to belong to the training set or to the training distribution. □

Corollary 1 (Approximation of the retraining displacement)**.** *Under the assumptions of the theorem,*

$$\left|n\left\|\hat{\theta}_{n,+z_\star} - \hat{\theta}_n\right\| - \left\|H_n^{-1}\nabla\ell(z_\star, \hat{\theta}_n)\right\|\right| = O(n^{-1}).$$

Thus, for a fixed training-set size, the norm of the parameter influence is a first-order-equivalent ranking score for the exact parameter displacement caused by adding the query sample.

B Appendix B : Supplementary Materials

B.1 Additional Experiments

Table 5: **Frozen backbone OOD detection results (AUROC ↑ / FPR95 ↓).** We use a ViT (Vision Transformer) pretrained on ImageNet-1k and fine-tune it on TinyImageNet by freezing all layers except the final classification layer. The in-distribution (ID) dataset is TinyImageNet. All experiments are repeated over 5 seeds and we report the averaged results. Best results in **bold**.

	SVHN		Textures		CIFAR-10		CIFAR-100	
Method	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓
MSP	$0.936_{\pm 0.007}$	$0.351_{\pm 0.023}$	$0.878_{\pm 0.031}$	$0.511_{\pm 0.052}$	$0.842_{\pm 0.009}$	$0.651_{\pm 0.015}$	$0.830_{\pm 0.005}$	$0.639_{\pm 0.004}$
Energy	$\mathbf{0.949}_{\pm 0.005}$	$0.263_{\pm 0.024}$	$0.880_{\pm 0.016}$	$0.508_{\pm 0.067}$	$0.850_{\pm 0.006}$	$0.572_{\pm 0.015}$	$0.834_{\pm 0.003}$	$0.599_{\pm 0.016}$
ODIN	$0.916_{\pm 0.006}$	$0.406_{\pm 0.024}$	$\mathbf{0.901}_{\pm 0.011}$	$0.404_{\pm 0.044}$	$0.814_{\pm 0.006}$	$0.675_{\pm 0.012}$	$0.807_{\pm 0.002}$	$0.658_{\pm 0.002}$
ASH	$0.948_{\pm 0.007}$	$0.302_{\pm 0.011}$	$0.868_{\pm 0.031}$	$0.524_{\pm 0.035}$	$0.839_{\pm 0.004}$	$0.601_{\pm 0.010}$	$0.827_{\pm 0.003}$	$0.615_{\pm 0.005}$
SCALE	$0.940_{\pm 0.007}$	$0.312_{\pm 0.026}$	$0.894_{\pm 0.016}$	$0.455_{\pm 0.066}$	$\mathbf{0.855}_{\pm 0.005}$	$0.586_{\pm 0.017}$	$0.838_{\pm 0.002}$	$0.598_{\pm 0.013}$
GradNorm	$0.934_{\pm 0.004}$	$0.331_{\pm 0.025}$	$0.884_{\pm 0.022}$	$0.491_{\pm 0.056}$	$0.800_{\pm 0.003}$	$0.564_{\pm 0.007}$	$0.803_{\pm 0.001}$	$0.585_{\pm 0.003}$
DataInf w/ pseudo	$0.935_{\pm 0.006}$	$0.276_{\pm 0.023}$	$0.882_{\pm 0.024}$	$0.424_{\pm 0.062}$	$0.826_{\pm 0.010}$	$0.498_{\pm 0.023}$	$0.838_{\pm 0.010}$	$0.596_{\pm 0.011}$
WIF (ours)	$0.935_{\pm 0.006}$	$\mathbf{0.250}_{\pm 0.025}$	$0.884_{\pm 0.023}$	$\mathbf{0.401}_{\pm 0.061}$	$0.826_{\pm 0.010}$	$\mathbf{0.493}_{\pm 0.022}$	$\mathbf{0.840}_{\pm 0.009}$	$\mathbf{0.578}_{\pm 0.010}$

B.2 Push-through identity trick

Lemma 1 (Push-through identity)**.** *For any $G \in \mathbb{R}^{d\times n}$ and any $\lambda > 0$,*

$$\left(\lambda I_d + GG^\top\right)^{-1} G \;=\; G\left(\lambda I_n + G^\top G\right)^{-1}. \tag{13}$$

Proof. Start from the trivial identity

$$G\big(\lambda I_n + G^\top G\big) \;=\; \lambda G + GG^\top G \;=\; \big(\lambda I_d + GG^\top\big)G.$$

Since $\lambda > 0$ we have $\lambda I_n + G^\top G \succeq \lambda I_n \succ 0$ and $\lambda I_d + GG^\top \succeq \lambda I_d \succ 0$, so both factors are invertible. Multiplying on the left by $(\lambda I_d + GG^\top)^{-1}$ and on the right by $(\lambda I_n + G^\top G)^{-1}$ yields equation 13. □

Corollary 2 (Dual form of the Woodbury step)**.** *Let*

$$M \;:=\; \left(I_n + \tfrac{1}{\lambda} G^\top G\right)^{-1} \;=\; \lambda\big(\lambda I_n + G^\top G\big)^{-1}, \qquad M' \;:=\; \left(I_d + \tfrac{1}{\lambda} GG^\top\right)^{-1} \;=\; \lambda\big(\lambda I_d + GG^\top\big)^{-1}.$$

Then

$$G\,M\,G^\top \;=\; M'\,GG^\top \;=\; GG^\top M', \tag{14}$$

and consequently

$$\left(GG^\top + \lambda I_d\right)^{-1} \;=\; \tfrac{1}{\lambda} I_d - \tfrac{1}{\lambda^2}\, G\,M\,G^\top \;=\; \tfrac{1}{\lambda} I_d - \tfrac{1}{\lambda^2}\, GG^\top M'. \tag{15}$$

That is, inverting the $n \times n$ matrix M and inverting the $d \times d$ matrix M' produce algebraically identical *operators: the choice between them is a matter of computational cost only, and introduces no approximation error.*

Proof. By Lemma 1,

$$G\,M\,G^\top = \lambda\left[\left(\lambda I_n + G^\top G\right)^{-1} \text{ pushed through}\right] = \lambda\left(\lambda I_d + GG^\top\right)^{-1} GG^\top = M'\,GG^\top,$$

where the middle step applies equation 13 to $G(\lambda I_n + G^\top G)^{-1} G^\top$. The second equality in equation 14 holds because M' is a rational function of $GG^\top$ and therefore commutes with it. Substituting equation 14 into Eq. (8) gives equation 15. □

Remark 1 (Direct verification of equation 15)**.** *The right-hand form can also be checked in one line without invoking Eq. (8). Since $\frac{1}{\lambda^2} GG^\top M' = \frac{1}{\lambda} GG^\top (\lambda I_d + GG^\top)^{-1}$,*

$$\tfrac{1}{\lambda} I_d - \tfrac{1}{\lambda^2} GG^\top M' = \tfrac{1}{\lambda}\left[I_d - GG^\top\left(\lambda I_d + GG^\top\right)^{-1}\right] = \tfrac{1}{\lambda}\left[\left(\lambda I_d + GG^\top\right) - GG^\top\right]\left(\lambda I_d + GG^\top\right)^{-1} = \left(\lambda I_d + GG^\top\right)^{-1}.$$

C ADDITIONAL

Broader Impacts. WIF aims to improve the reliability of deployed machine learning systems by allowing them to flag unfamiliar inputs before acting on them, which is directly relevant to safety-critical settings such as medical diagnosis, autonomous driving, and large-scale content moderation where silent failure on out-of-distribution inputs can cause real harm. We do not anticipate any direct negative societal impact specific to WIF, as the method is a post-hoc scoring rule that operates on top of existing trained models and does not generate or disseminate content. As with any OOD detection method, however, deployers should be aware that imperfect scoring can yield both false acceptances of unfamiliar inputs and false rejections of valid ones, and the calibration of decision thresholds in high-stakes applications remains the responsibility of the deployer.

Limitations. We discuss several limitations of WIF that point to directions for future work. First, the approximation-quality validation in Sec. 5.1 is conducted on a CNN trained on MNIST because the ground-truth reference requires exactly retraining the model on $\mathcal{D} \cup \{z_\star\}$ for each query sample, which becomes prohibitively expensive on larger architectures. We complement this tractable setup with a LiSSA-reference comparison on ResNet-18/CIFAR-10 in the same section, but a direct retraining-based validation on deeper backbones remains out of reach with current compute budgets.

Second, our experiments are limited to image classification with backbones up to ResNet-18 and ViT. Whether WIF retains its faithfulness and downstream gains on substantially larger architectures (e.g., large language models) or on other modalities such as generative models and sequence models is left to future work. The layer-wise block-diagonal approximation underlying Eq. 3 also discards cross-layer Hessian terms, and although this is a standard approximation in the IF literature, its effect may become more pronounced as depth and width grow.

Third, we did not perform a systematic sensitivity analysis of the damping coefficient λ, which controls the trade-off between numerical stability and approximation fidelity in Eq. 8. In our experiments λ was chosen on a held-out subset, and we observed stable behavior across the values tried; nevertheless, a principled study of how λ should scale with dataset size, network width, and the ID/OOD gap would strengthen the practical guidance for deploying WIF on new domains.

Finally, the pseudo-labeling component (Eq. 7) inherits the confidence calibration of the underlying classifier. On models with severe miscalibration or systematic prediction bias on near-OOD inputs, the alignment effect identified in Sec. 5.2 may weaken, and pairing WIF with explicit calibration techniques is a natural extension we leave open.

Implementation Details. All experiments are conducted on a single NVIDIA RTX 4060 GPU. Backbones are trained with the Adam optimizer at a learning rate of 1×10^{-4}. The damping coefficient λ in Eq. 5 is selected from $\{0.01, 0.001\}$ on a held-out subset for each backbone, and we observed stable behavior across both values.

D ASSET LICENSES

Table 6: Datasets and code assets used in this paper.

Asset	Type	License
MNIST	Dataset	CC BY-SA 3.0
CIFAR-10/100	Dataset	MIT-style (research use)
SVHN	Dataset	Non-commercial research use
Textures (DTD)	Dataset	Research use
Tiny ImageNet	Dataset	ImageNet terms of use
PyTorch	Framework	BSD-3-Clause

All assets are used in accordance with their respective licenses and terms of use. Baseline scoring rules (MSP, Energy, ODIN, ASH, SCALE, GradNorm) and continual-learning methods (EWC, GEM)

are reimplemented following the descriptions in the cited papers; their original code repositories, where applicable, are released under permissive open-source licenses (typically MIT or Apache 2.0).