GROUPS DON'T MOVE AS ONE: A BREGMAN CLUSTERING VIEW OF LINEARIZED GROUP INFLUENCE

Anonymous authors
Paper under double-blind review

ABSTRACT

Influence functions provide an efficient alternative to retraining for assessing the effects of removing or reweighting training data. For groups, however, the standard first-order estimate reduces individual contributions to a sum that can conceal substantial differences among members, including opposing effects that cancel. We establish a Bregman centroid formulation of the linearized group influence objective, interpreting this aggregate as a single representative and using Bregman information (BI) to quantify the variation it omits. Extending this formulation to multiple clusters reveals distinct contributions whose sum recovers the original estimate. We further propose an estimator that jointly optimizes the weights of these cluster contributions, rather than simply summing them. Experiments uncover cluster-specific influence patterns and a positive association between target-specific BI and approximation error. The proposed estimator achieves higher R^2 than standard group influence in predicting target changes under the nonlinear proximal Bregman response function. Simplified clustering and correction variants reduce computational cost. The centroid view thus explains what additive group influence leaves out and provides a basis for using that information to improve estimation.

1 INTRODUCTION

Influence functions estimate how removing or reweighting training data affects a trained model without retraining it for every perturbation (Jaeckel, 1972; Hampel, 1974; Cook, 1977; Koh & Liang, 2017). By approximating changes in model parameters and quantities such as validation loss, they help us understand how training data contribute to model behavior. Group influence extends this analysis from individual examples to subsets of training data, estimating the joint effect of removing or reweighting their members (Basu et al., 2020; Koh et al., 2019; Heo et al., 2026).

Standard first-order group influence estimates this joint effect by summing individual influence estimates (Koh et al., 2019). However, the sum hides differences among group members (Basu et al., 2020; Heo et al., 2026). For example, a near-zero estimated change in validation loss may reflect uniformly weak individual contributions or large opposing contributions that cancel. We seek to characterize this hidden structure and investigate whether accounting for it can improve group influence estimation.

We show that the linearized objective underlying group influence is equivalent to a Bregman centroid problem (Banerjee et al., 2005; Nielsen & Nock, 2008), providing an alternative perspective grounded in Bregman geometry. In this formulation, the group estimate is a single representative of scaled individual parameter-change estimates. This connection identifies both what aggregation retains, their center, and what it hides, their variation around that center. Bregman information (BI) provides a measure of this hidden variation.

We extend this view from one centroid to several through Bregman clustering (Banerjee et al., 2005). The resulting clusters reveal how different parts of the group reinforce or oppose one another, while their contributions still sum to the original group estimate. Building on this decomposition, we propose an estimator that assigns a separate weight to each cluster's contribution. We jointly optimize these weights using a refined approximation of how the group affects the model. This keeps the centroids fixed while allowing the relative contributions of different clusters to change.

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.

Our experiments reveal distinct cluster-wise influence patterns hidden by the aggregate and show a positive association between target-specific BI and approximation error. The proposed estimator achieves higher R^2 than standard group influence when evaluated against target changes under the nonlinear proximal Bregman response function (PBRF) (Bae et al., 2022). We also examine simpler clustering and correction schemes that reduce computation. Together, these findings show that examining within-group structure can support both a finer understanding of group influence and more accurate estimation.

2 PRELIMINARIES

We introduce additive group influence and the proximal Bregman response function it approximates. Related work is discussed in Appendix A.

2.1 INFLUENCE FUNCTIONS AND ADDITIVE GROUP INFLUENCE

Let $\mathcal{D}_{\text{train}} = \{(x_i, y_i)\}_{i=1}^N$ be the training set and h_θ a model with parameters $\theta \in \mathbb{R}^d$. The per-example loss $\ell(u, y)$ is twice differentiable and convex in the model output u. We write $L_i(\theta) \coloneqq \ell(h_\theta(x_i), y_i)$ and $\mathcal{J}(\theta) \coloneqq N^{-1}\sum_{i=1}^N L_i(\theta)$. The *original model*, trained before any data removal or reweighting, has parameters θ_0. All gradients $z_i \coloneqq \nabla_\theta L_i(\theta_0)$ and curvature matrices below are evaluated at these parameters.

For a nonempty group $\mathcal{S} \subseteq [N]$, uniform reweighting gives

$$\theta^\star_{\mathcal{S}}(\epsilon) \coloneqq \arg\min_\theta \left\{ \mathcal{J}(\theta) + \epsilon \sum_{i\in\mathcal{S}} L_i(\theta) \right\}.$$

We call $\epsilon \sum_{i\in\mathcal{S}} L_i(\theta)$ the *group-reweighting term*. It changes the weights of the group's existing training losses rather than introducing a separate loss function. We refer to the resulting change in model parameters, $\theta^\star_{\mathcal{S}}(\epsilon) - \theta_0$, as the model's *parameter response* to this group reweighting. Setting $\epsilon = -1/N$ removes the group's loss terms, with $|\mathcal{S}| = 1$ recovering leave-one-out removal.

Under the classical regularity conditions, influence functions approximate this parameter response to first order in ϵ (Hampel, 1974; Cook & Weisberg, 1982; Koh & Liang, 2017):

$$\theta^\star_{\mathcal{S}}(\epsilon) - \theta_0 \;\approx\; -\epsilon H^{-1} \sum_{i\in\mathcal{S}} z_i, \qquad H \coloneqq \nabla^2_\theta \mathcal{J}(\theta_0). \tag{1}$$

The derivation and its regularity conditions are given in Appendix B.1.

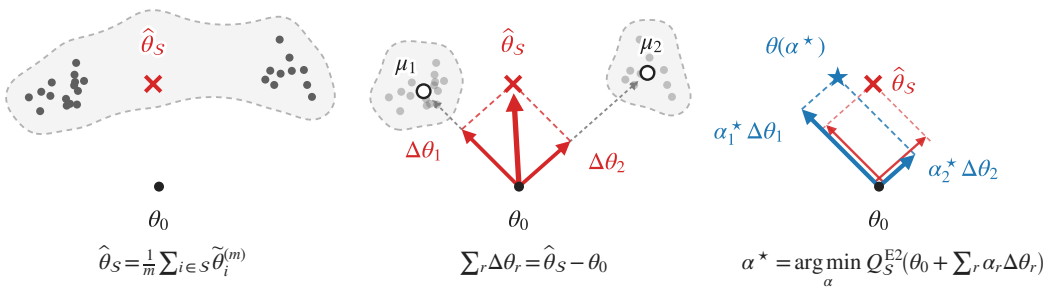
For neural networks, H can be singular or indefinite (Bae et al., 2022). We focus on the estimate obtained by replacing it with the damped generalized Gauss-Newton (GGN) matrix (Schraudolph, 2002; Martens, 2010; Botev et al., 2017)

$$G \coloneqq \frac{1}{N}\sum_{v=1}^N J_v^\top H_v J_v + \lambda I_d,$$

where J_v is the Jacobian of $h_\theta(x_v)$ and H_v is the Hessian of $\ell(\cdot, y_v)$ at $h_{\theta_0}(x_v)$. Convexity of the loss in its output and $\lambda > 0$ ensure $G \succ 0$ (Martens, 2010). The resulting group estimate, denoted by $\widehat{\theta}_{\mathcal{S}}(\epsilon)$, is additive because the same inverse-curvature matrix acts on every member gradient (Koh et al., 2019):

$$\begin{aligned} \widehat{\theta}_{\mathcal{S}}(\epsilon) - \theta_0 &= -\epsilon G^{-1} \sum_{i\in\mathcal{S}} z_i \\ &= \sum_{i\in\mathcal{S}} \big(\widehat{\theta}_{\{i\}}(\epsilon) - \theta_0\big), \qquad \widehat{\theta}_{\{i\}}(\epsilon) = \theta_0 - \epsilon G^{-1} z_i. \end{aligned} \tag{2}$$

For a differentiable scalar target f, we seek to estimate how its value changes under training-group reweighting. For example, a held-out example (x', y'), which we call a *query*, defines the target $f(\theta) = \ell(h_\theta(x'), y')$ (Koh & Liang, 2017; Heo et al., 2026). Our formulation also allows aggregate



(a) Centroid representation (b) Component decomposition (c) Component reweighting

Figure 1: Illustration of the Bregman clustering view of linearized group influence. **(a)** Standard GI summarizes member-wise response points by a single centroid, hiding within-group structure. **(b)** Bregman clustering reveals distinct component contributions while preserving the original aggregate estimate. **(c)** Joint E2 reweights these contributions under a corrected objective, yielding a refined estimate (blue star).

targets, such as the mean loss on a held-out set. Standard group influence (GI) uses the estimated parameter response to predict the corresponding target change:

$$\widehat{A}_{\mathcal{S}}(\epsilon) := \nabla_\theta f(\theta_0)^\top \big(\widehat{\theta}_{\mathcal{S}}(\epsilon) - \theta_0\big) = -\epsilon \sum_{i\in\mathcal{S}} q^\top z_i, \qquad q := G^{-1}\nabla_\theta f(\theta_0). \tag{3}$$

We refer to q as the *target direction* associated with f.

2.2 PBRF and Its Linearized Surrogate

Bae et al. (2022) show that damped-GGN influence estimates approximate the parameter response defined by the proximal Bregman response function (PBRF) to first order in the reweighting strength. Writing this construction for a reweighted group $\mathcal{S}$ gives the objective

$$Q_{\mathcal{S}}^{\mathrm{PBRF}}(\theta;\epsilon) := \frac{1}{N}\sum_{v=1}^{N} D_{\ell(\cdot,y_v)}\big(h_\theta(x_v), h_{\theta_0}(x_v)\big) + \epsilon \sum_{i\in\mathcal{S}} L_i(\theta) + \frac{\lambda}{2}\|\theta-\theta_0\|_2^2, \tag{4}$$

where $D_\phi(u,v) := \phi(u) - \phi(v) - \nabla\phi(v)^\top(u-v)$ is the Bregman divergence generated by a differentiable convex function ϕ (Bregman, 1967; Banerjee et al., 2005). The first and third terms penalize changes in model outputs and parameters, respectively, while the second is the group-reweighting term introduced in Section 2.1. Following this characterization, we use the nonlinear PBRF response as our evaluation reference, rather than the response to the reweighted training objective.

Since θ_0 minimizes the PBRF objective at $\epsilon = 0$, we use a local surrogate around θ_0. Dropping the parameter-independent constant $\epsilon\sum_{i\in\mathcal{S}} L_i(\theta_0)$ gives the *linearized group PBRF objective*:

$$Q_{\mathcal{S}}^{\mathrm{lin}}(\theta;\epsilon) := D_{\Phi_G}(\theta,\theta_0) + \epsilon c_{\mathcal{S}}^\top(\theta-\theta_0), \qquad \Phi_G(\theta) := \tfrac{1}{2}\theta^\top G\theta, \quad c_{\mathcal{S}} := \sum_{i\in\mathcal{S}} z_i. \tag{5}$$

Here $D_{\Phi_G}(\theta,\theta') = \frac{1}{2}(\theta-\theta')^\top G(\theta-\theta')$. Since $G \succ 0$, the unique minimizer is $\widehat{\theta}_{\mathcal{S}}(\epsilon) = \theta_0 - \epsilon G^{-1}c_{\mathcal{S}}$, recovering Equation (2). The second-order expansion of the first term, together with the third term, has curvature G at θ_0. Thus, D_{Φ_G} expresses this local penalty in parameter space, whereas $D_{\ell(\cdot,y_v)}$ in Equation (4) acts in model-output space. The derivation and the local regularity conditions relating this estimate to the first-order PBRF response are given in Appendix B.2.

3 A Bregman Clustering View of Group Influence

The additive estimate in Equation (2) describes the group's overall response but not the variation among its members, as illustrated in Figure 1(a). We characterize this aggregation by showing that the linearized group PBRF objective is a Bregman centroid objective. This identifies both the single representative retained by GI and the dispersion it omits, and leads to a multi-component representation that preserves the original estimate, as illustrated in Figure 1(b).

3.1 Group Influence as a Bregman Centroid

Let $m := |\mathcal{S}|$. To expose the member-wise structure of Equation (5), first write it as an average:

$$Q_{\mathcal{S}}^{\mathrm{lin}}(\theta;\epsilon) = \frac{1}{m}\sum_{i\in\mathcal{S}}\left[D_{\Phi_G}(\theta,\theta_0) + \epsilon m z_i^\top(\theta-\theta_0)\right]. \tag{6}$$

Each summand has a unique minimizer,

$$\begin{aligned}\widetilde{\theta}_i^{(m)}(\epsilon) &:= \arg\min_{\vartheta\in\mathbb{R}^d}\left\{D_{\Phi_G}(\vartheta,\theta_0) + \epsilon m z_i^\top(\vartheta-\theta_0)\right\}\\ &= \theta_0 - \epsilon m G^{-1} z_i.\end{aligned} \tag{7}$$

We call these minimizers *member-wise response points* and collect them in the response cloud $\mathcal{P}_{\mathcal{S}}(\epsilon) := \{\widetilde{\theta}_i^{(m)}(\epsilon) : i \in \mathcal{S}\}$. Relative to the singleton estimate in Equation (2), each point scales the estimated parameter displacement by m.

A Bregman centroid minimizes the average divergence from a collection of points to one representative (Banerjee et al., 2005). The next theorem identifies the entire linearized group objective with such a centroid objective.

Theorem 3.1 (Bregman centroid representation of the linearized group PBRF)**.** *Let $G \succ 0$, let Φ_G and $Q_{\mathcal{S}}^{\mathrm{lin}}$ be as in Equation (5), and let $\mathcal{S} \subseteq [N]$ have size m with response points defined by Equation (7). For every $\theta \in \mathbb{R}^d$,*

$$Q_{\mathcal{S}}^{\mathrm{lin}}(\theta;\epsilon) = \frac{1}{m}\sum_{i\in\mathcal{S}} D_{\Phi_G}\left(\widetilde{\theta}_i^{(m)}(\epsilon),\theta\right) + K_{\mathcal{S}}(\epsilon), \tag{8}$$

where $K_{\mathcal{S}}(\epsilon) := -\frac{\epsilon^2 m}{2}\sum_{i\in\mathcal{S}} z_i^\top G^{-1} z_i$ is independent of θ. The unique minimizer is

$$\widehat{\theta}_{\mathcal{S}}(\epsilon) \;=\; \frac{1}{m}\sum_{i\in\mathcal{S}}\widetilde{\theta}_i^{(m)}(\epsilon) \;=\; \theta_0 - \epsilon G^{-1} c_{\mathcal{S}}. \tag{9}$$

The proof is given in Appendix B.3. Appendix B.6 gives a dual-coordinate extension to general Legendre generators and states where the quadratic specialization is needed. The result is an objective identity for every θ, not merely an equality of minimizers. It shows that standard GI fits a single representative to the response cloud in the geometry induced by G. The centroid objective also provides a way to quantify the variation hidden by this representative.

3.2 Bregman Information of the Response Cloud

Groups with the same centroid can have different member-wise responses. For a scalar target, for example, a small aggregate can arise from uniformly weak contributions or from large opposing contributions that cancel. *Bregman information* (BI) measures this omitted dispersion as the minimum average divergence from the response points to a single representative (Banerjee et al., 2005), corresponding to the minimum value of the first term on the right-hand side of Equation (8).

We first measure dispersion in parameter space, without reference to a particular target. The minimum is attained at the group centroid $\widehat{\theta}_{\mathcal{S}}(\epsilon)$. Because each member-wise displacement in Equation (7) is scaled by ϵm, the resulting quadratic divergence contains a common factor $\epsilon^2 m^2$. We separate this explicit scaling from the within-group dispersion by writing

$$\frac{1}{m}\sum_{i\in\mathcal{S}} D_{\Phi_G}\left(\widetilde{\theta}_i^{(m)}(\epsilon),\widehat{\theta}_{\mathcal{S}}(\epsilon)\right) = \epsilon^2 m^2 \mathcal{B}_G(\mathcal{S}),$$

where the *normalized full Bregman information* is

$$\mathcal{B}_G(\mathcal{S}) := \frac{1}{2m}\sum_{i\in\mathcal{S}}(z_i-\bar{z}_{\mathcal{S}})^\top G^{-1}(z_i-\bar{z}_{\mathcal{S}}), \qquad \bar{z}_{\mathcal{S}} := \frac{1}{m}\sum_{i\in\mathcal{S}} z_i. \tag{10}$$

To measure how each response differs in their effects on a scalar target f, we instead consider their first-order target changes:

$$f\left(\widetilde{\theta}_i^{(m)}(\epsilon)\right) - f(\theta_0) \;\approx\; \nabla_\theta f(\theta_0)^\top\left(\widetilde{\theta}_i^{(m)}(\epsilon)-\theta_0\right) \;=\; -\epsilon m q^\top z_i, \tag{11}$$

where $q = G^{-1}\nabla_\theta f(\theta_0)$ is the target direction defined in Equation (3). The mean of these linearized target changes recovers standard GI, while their dispersion describes variation hidden by this mean. We measure dispersion among these linearized target changes using half the squared difference, $D_\psi(a,b) = \frac{1}{2}(a-b)^2$, which is the Bregman divergence generated by $\psi(a) = a^2/2$. The optimal single representative under this divergence is the mean, and the minimum average divergence is therefore half the variance of the linearized target changes. Removing the common factor $\epsilon^2 m^2$ gives the *normalized projected Bregman information*:

$$\mathcal{B}_{\mathrm{proj}}(\mathcal{S}) \coloneqq \frac{1}{2m}\sum_{i\in\mathcal{S}}\left[q^\top(z_i - \bar{z}_\mathcal{S})\right]^2 = \frac{1}{2}\,\mathrm{Var}_{i\sim\mathrm{Unif}(\mathcal{S})}(q^\top z_i). \tag{12}$$

Projected BI therefore measures variation among the unscaled influence scores for the chosen target, rather than variation across the full parameter-response space. The normalization identities are proved in Appendix B.4.

Both quantities characterize dispersion omitted by the centroid; neither by itself bounds the error relative to the nonlinear PBRF. We examine the association between projected BI and approximation error in Section 5.2.

3.3 From One Centroid to Multiple Components

To describe within-group structure beyond a single BI value, we partition $\mathcal{S}$ into k nonempty, disjoint components $\mathcal{C}_1, \ldots, \mathcal{C}_k$, for a chosen $k \in \{1, \ldots, m\}$. Each $\mathcal{C}_r$ indexes a subset of the response cloud, with representative $\mu_r \in \mathbb{R}^d$. Bregman clustering jointly selects the partition and representatives by minimizing the following objective (Banerjee et al., 2005):

$$\min_{\substack{\mathcal{S}=\bigsqcup_{r=1}^k \mathcal{C}_r,\ \mathcal{C}_r\neq\varnothing \\ \mu_1,\ldots,\mu_k\in\mathbb{R}^d}} \frac{1}{m}\sum_{r=1}^{k}\sum_{i\in\mathcal{C}_r} D_{\Phi_G}\big(\widetilde{\theta}_i^{(m)}(\epsilon), \mu_r\big). \tag{13}$$

For a fixed partition, the optimal representative μ_r is the mean of the response points in $\mathcal{C}_r$. Its size-weighted displacement defines the *component contribution*

$$\Delta\theta_r \coloneqq \frac{|\mathcal{C}_r|}{m}(\mu_r - \theta_0) = -\epsilon G^{-1}\sum_{i\in\mathcal{C}_r} z_i. \tag{14}$$

Using the same G and ϵ for all components, any partition satisfies

$$\sum_{r=1}^{k}\Delta\theta_r = \widehat{\theta}_\mathcal{S}(\epsilon) - \theta_0. \tag{15}$$

The target contributions $\nabla_\theta f(\theta_0)^\top \Delta\theta_r$ therefore also sum to standard GI. The decomposition therefore reveals structure without changing the additive estimate.

4 Improving Group Influence with Component Structure

The component decomposition in Equation (15) expresses the standard parameter-response estimate as a sum of component contributions with unit coefficients. We keep these component directions fixed and optimize their coefficients using a richer approximation of the group-reweighting term. The resulting parameter change remains in the span of these directions but is no longer restricted to their original sum, as illustrated in Figure 1(c).

4.1 Quadratic Correction in Component Space

We parameterize the group-induced change by

$$\theta(\alpha) = \theta_0 + \sum_{r=1}^{k}\alpha_r\Delta\theta_r, \qquad \alpha\in\mathbb{R}^k. \tag{16}$$

Here $\theta(\mathbf{1}) = \widehat{\theta}_{\mathcal{S}}(\epsilon)$ is already the unique minimizer of Equation (5). Obtaining a different parameter response therefore requires modifying the objective used to optimize α.

In Equation (5), the group-reweighting term $\epsilon c_{\mathcal{S}}^{\top}(\theta - \theta_0)$ is linear in the parameter displacement and therefore omits the curvature of the reweighted training losses. We incorporate this curvature through a GGN-based quadratic approximation, while leaving the common penalty $D_{\Phi_G}(\theta, \theta_0)$ unchanged:

$$Q_{\mathcal{S}}^{\mathrm{E2}}(\theta; \epsilon) \coloneqq Q_{\mathcal{S}}^{\mathrm{lin}}(\theta; \epsilon) + \frac{\epsilon}{2} \sum_{i \in \mathcal{S}} (\theta - \theta_0)^{\top} J_i^{\top} H_i J_i (\theta - \theta_0). \tag{17}$$

Here J_i and H_i are evaluated at θ_0, as in Section 2.1, and removal uses $\epsilon = -1/N$. We refer to this correction as E2. After substituting Equation (16), the added term includes curvature along each component direction and cross terms between directions, evaluated over the entire reweighted group.

In *Joint E2*, we jointly optimize the component coefficients by minimizing $Q_{\mathcal{S}}^{\mathrm{E2}}$:

$$\alpha^{\star} \in \arg\min_{\alpha \in \mathbb{R}^k} Q_{\mathcal{S}}^{\mathrm{E2}}\big(\theta(\alpha); \epsilon\big). \tag{18}$$

The resulting target-change prediction is $\nabla_{\theta} f(\theta_0)^{\top}(\theta(\alpha^{\star}) - \theta_0)$. For $k = 1$, coefficient fitting can only rescale the original group direction; multiple components allow their relative contributions to change.

4.2 Efficient Approximations

Joint E2 requires both component construction and evaluation of the GGN curvature of the reweighted group. We consider two approximations that reduce these costs separately.

Within-component curvature approximation. For each reweighted example, Joint E2 evaluates the GGN-based quadratic correction using the combined displacement $\theta(\alpha) - \theta_0 = \sum_{r=1}^{k} \alpha_r \Delta\theta_r$. For an example in $\mathcal{C}_r$, *Intra E2* instead uses only $\alpha_r \Delta\theta_r$ in this additional quadratic term, while leaving $Q_{\mathcal{S}}^{\mathrm{lin}}(\theta(\alpha); \epsilon)$ unchanged:

$$Q_{\mathcal{S}}^{\mathrm{intra}}(\alpha; \epsilon) \coloneqq Q_{\mathcal{S}}^{\mathrm{lin}}\big(\theta(\alpha); \epsilon\big) + \frac{\epsilon}{2} \sum_{r=1}^{k} \alpha_r^2 \sum_{i \in \mathcal{C}_r} \Delta\theta_r^{\top} J_i^{\top} H_i J_i \Delta\theta_r. \tag{19}$$

This restriction omits cross-direction terms and evaluates each direction's curvature only on its own component's examples. The unchanged common quadratic term retains cross-component terms, so we still optimize all coefficients jointly.

Euclidean component construction. To reduce clustering cost in Equation (13), we form components by clustering the raw gradients z_i using Euclidean distances, without curvature weighting. This simplification applies only to clustering; we retain the original G when computing component directions and optimizing their coefficients.

5 Experiments

We investigate whether projected BI is associated with the approximation error of standard GI and whether the proposed component-based correction improves estimation accuracy. We first describe the common experimental setup in Section 5.1 and then examine the BI-error relationship in Section 5.2. Finally, we analyze component-wise influence patterns and evaluate the proposed corrections in Section 5.3.

5.1 Experimental Setup

Models and removal groups. We use logistic regression on Dogfish (Koh & Liang, 2017; Koh et al., 2019), a multilayer perceptron (MLP) on MNIST (Bae et al., 2022), and ResNet9 on CIFAR-10 (Park et al., 2023). For each dataset, we fix the original model θ_0 and select 100 held-out test examples as *queries*, as defined in Section 2.1. For each removal ratio, we sample 100 training groups uniformly without replacement within each group, allowing overlap between groups. We predict how each group's removal changes the loss on each individual query. All methods use the same removal groups and queries.

Targets and reference responses. Query (x'_j, y'_j) defines the single-example loss $f_j(\theta) := \ell(h_\theta(x'_j), y'_j)$. For a fixed query j, we write $f = f_j$ and suppress the query index. With removal weight $\epsilon = -1/N$, we numerically optimize the PBRF objective in Equation (4) once per group and reuse the resulting parameters across methods and queries. The reference loss change for the query under evaluation is

$$A_{\mathcal{S}}^{\mathrm{PBRF}}(\epsilon) := f\big(\theta_{\mathcal{S}}^{\mathrm{PBRF}}(\epsilon)\big) - f(\theta_0).$$

We use the full-training-set damped GGN matrix G, with the same ridge coefficient in G and the PBRF proximal term.

Evaluation metrics. For each method and query, we compute Spearman correlation and raw R^2 from 100 pairs of predicted and reference loss changes, one per removal group. We repeat this evaluation for the 100 queries and report the median of each metric over the common retained subset. High rank correlation can coexist with large prediction errors (Koh et al., 2019). We therefore use raw R^2 alongside Spearman to assess how closely predicted influence values match the reference, accounting for scale differences across queries. Evaluation details are provided in Appendix C.

5.2 Bregman Information and Approximation Error

BI characterizes dispersion omitted by the centroid representation, but its association with nonlinear approximation error is an empirical question. Using the 1% groups for all three models, we compute, for each fixed query, the Spearman correlation across removal groups between projected BI and the squared error of standard GI,

$$\mathcal{E}_{\mathcal{S}}(f, \epsilon) := \big[A_{\mathcal{S}}^{\mathrm{PBRF}}(\epsilon) - \widehat{A}_{\mathcal{S}}(\epsilon)\big]^2, \qquad (20)$$

where $\widehat{A}_{\mathcal{S}}(\epsilon)$ is defined in Equation (3). We plot the summed quantity $m\mathcal{B}_{\mathrm{proj}}(\mathcal{S})$ without the removal factor N^{-2}, following the convention detailed in Appendix C.3. Since the group size is fixed within each model, this rescaling does not affect the Spearman correlation. For each model, we display the query whose correlation is closest to the median across the 100 fixed queries. Figure 2 shows a positive rank association between projected BI and squared GI error in all three settings. This result empirically links the dispersion omitted by the centroid representation to GI approximation error.

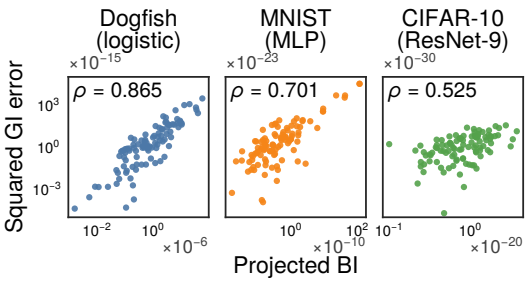


Figure 2: Bregman information tracks group influence error. Each point represents a random 1% removal group (100 groups per model). We plot projected Bregman information against the squared error of GI relative to the actual PBRF change. For each model, we show the query whose Spearman correlation ρ is closest to the median over 100 test queries.

Table 8 compares projected BI with full BI and Euclidean gradient variance. Projected BI exhibits the strongest correlation with squared GI error, whereas full BI and Euclidean gradient variance perform similarly. This gap shows that target projection provides additional information for the target-specific approximation error considered here. The comparison concerns dispersion measures; we evaluate the effect of clustering geometry on correction accuracy separately in Section 5.3.

5.3 Component Decomposition and Influence Correction

We first examine the component structure hidden by GI, then evaluate whether reweighting these components improves prediction and how the efficient approximations affect accuracy and cost.

Component-wise influence patterns. We partition each of the 100 random MNIST 10% removal groups into $k = 2$ Bregman components. For each of the 100 fixed queries, we compare the component contributions with their sum, which recovers GI by Equation (15). Component contributions are totals over the training examples in each component, rather than per-example averages. To examine class-wise patterns, we average each component's influence over test examples within each class in the 10,000-example MNIST test set, forming a ten-dimensional profile.

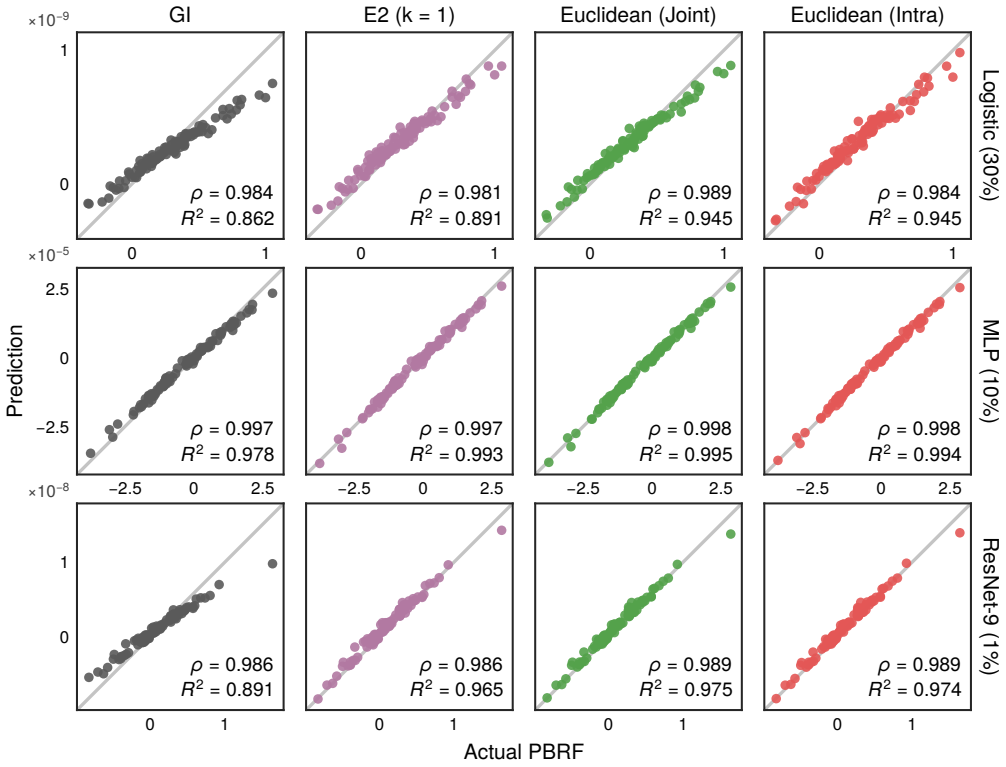


Figure 3: Actual PBRF vs. predicted test-loss changes. From top to bottom, the rows show Dogfish logistic regression (30% removal), MNIST MLP (10%), and CIFAR-10 ResNet9 (1%). From left to right, the columns compare GI, single-component E2 ($k = 1$), Euclidean Joint E2 ($k = 2$), and Euclidean Intra E2 ($k = 2$). Each panel shows one query across 100 removal groups, while annotations report median query-wise Spearman ρ and raw R^2 over the retained queries. The gray line marks $y = x$. Across all three models, the improvements of both $k = 2$ variants over single-component E2 are significant at the 0.001 level for both metrics after Holm correction. Statistical details are provided in Appendix C.3.

Figure 4(a) shows that the two components capture distinct class-wise influence patterns within a single removal group, with opposite-signed contributions for several classes and a cosine similarity of -0.46 between their unit-normalized profiles. Figure 4(b) shows that the component profiles are generally not aligned across the 100 removal groups, with a median cosine similarity of -0.03 and a 90th percentile of 0.36. The ablation in Appendix D.4 further shows that retaining both components yields higher median raw R^2 against the full-group PBRF reference than retaining either alone.

Accuracy of corrected estimates. Figure 3 compares GI, single-component E2 ($k = 1$), and Euclidean Joint and Intra E2 at $k = 2$ across the three models. Single-component E2 optimizes one coefficient along the original GI direction, whereas the two-component methods reweight the component contributions sep-

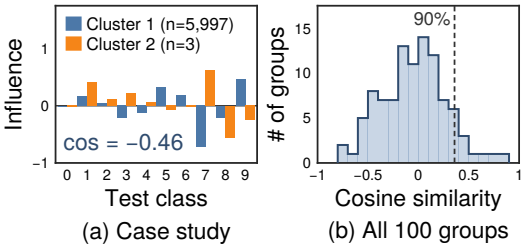


Figure 4: The two clusters tend to exhibit different class-wise influence patterns. For an MNIST MLP, we split each of 100 random 10% removal groups into $k = 2$ Bregman clusters. Each cluster's class-wise influence vector contains its contribution to GI averaged over test points within each class. **(a)** Case study: unit-normalized class-wise influence vectors of the two clusters. **(b)** Cosine similarities across all 100 groups (median -0.03, IQR $[-0.25, 0.17]$). The dashed line marks the 90th percentile (0.36).

Table 1: Ablation of clustering geometry and edit-curvature scope at $k = 2$. For each query, Spearman ρ and raw R^2 are computed across 100 random removal groups. Entries report the median over the common retained queries, with the interquartile range in brackets.

		Dogfish (logistic, 30%)		MNIST (MLP, 10%)	
	Clustering	Spearman $\rho \uparrow$	$R^2 \uparrow$	Spearman $\rho \uparrow$	$R^2 \uparrow$
Joint	Bregman	$0.989_{[0.982,\,0.991]}$	$0.945_{[0.926,\,0.957]}$	$0.998_{[0.997,\,0.998]}$	$0.995_{[0.993,\,0.996]}$
	Euclidean	$0.989_{[0.983,\,0.991]}$	$0.945_{[0.926,\,0.958]}$	$0.998_{[0.997,\,0.998]}$	$0.995_{[0.993,\,0.996]}$
Intra	Bregman	$0.984_{[0.977,\,0.989]}$	$0.945_{[0.919,\,0.957]}$	$0.998_{[0.997,\,0.998]}$	$0.993_{[0.991,\,0.995]}$
	Euclidean	$0.984_{[0.976,\,0.989]}$	$0.945_{[0.919,\,0.957]}$	$0.998_{[0.997,\,0.998]}$	$0.994_{[0.992,\,0.995]}$

arately. Although GI already ranks groups accurately, consistent with Koh et al. (2019), single-component E2 improves raw R^2 over GI on all three models. Joint E2 further raises median raw R^2 from 0.891 to 0.945 on Dogfish and from 0.965 to 0.975 on ResNet9; Intra E2 reaches 0.945 and 0.974, respectively. On MNIST, where single-component E2 already achieves 0.993, Joint and Intra E2 reach 0.995 and 0.994. The paired significance tests in Appendix C.3 show that both two-component variants improve query-wise Spearman correlation and raw R^2 over single-component E2 in all three settings, with $p_{\text{adj}} < 0.001$ throughout. Thus, separately reweighting the component contributions improves on rescaling the aggregate direction alone.

Efficient approximations. Table 1 separates the effects of clustering geometry and edit-curvature scope by comparing Bregman and Euclidean clustering under Joint and Intra E2 at $k = 2$ on Dogfish and MNIST. Changing the clustering geometry has little effect under either curvature scope, and Intra E2 remains close to Joint E2 for either geometry. The empirical evidence and attribution precedents supporting the Euclidean approximation are discussed in Appendix D.2.

Table 2: Measured stage-wise cost reductions on MNIST at $k = 2$. Baseline denotes Bregman clustering for partition construction and Joint E2 for edit curvature; Approx. denotes Euclidean clustering and Intra E2, respectively. Reduction is the relative decrease from the baseline stage time. Big-O analysis and timing protocols are provided in Appendix C.8.

Stage	Removal ratio	Baseline	Approx.	Reduction
Partition	1%	59.4s	0.277s	99.5%
Edit curvature	10%	0.193s	0.103s	46.6%

Table 2 reports the corresponding measured stage-wise cost reductions on MNIST: Euclidean clustering substantially reduces partition-construction time, while Intra E2 reduces edit-curvature time with fixed Euclidean partitions. These results identify computational savings in the replaced stages, alongside similar prediction accuracy in the evaluated $k = 2$ settings. Complexity analysis and timing details are provided in Appendix C.8.

6 Conclusion

We show that standard group influence compresses member-wise responses into a single representative, hiding variation within the group. The Bregman view makes this compression explicit: BI quantifies the omitted variation, while multiple components preserve the original estimate and reveal how different parts of the group reinforce or oppose one another. Experiments show that target-projected BI is associated with approximation error relative to the nonlinear PBRF, and that adjusting the component weights improves prediction of target changes. The Bregman characterization is exact for the linearized response, whereas the BI–error association and estimation gains are empirical. Together, these results show that structure hidden by additive aggregation can be exposed and used to refine group influence estimates.

AI USE STATEMENT

We used generative AI tools to support research ideation and execution, including refining experimental designs, implementing and debugging experimental and analysis code, and interpreting experimental results. We also used these tools to draft and revise sections of the manuscript and to improve the clarity and readability of the writing. The authors made the final decisions regarding the methodology, interpretation of results, and presentation, and take responsibility for the final content of this work, including all AI-assisted text, code, analyses, and other artifacts.

ETHICS STATEMENT

This work studies methods for interpreting and estimating the effect of training-data removal. The experiments use standard benchmark datasets and fixed research models, and do not involve human-subject experiments, newly collected personal data, or deployment in real-world decision systems. The proposed method can support beneficial uses such as data auditing, debugging, and understanding when additive group influence hides within-group structure. However, influence estimates are approximations and should not be treated as definitive evidence about individual examples, especially in sensitive or consequential settings. Practitioners should use appropriate privacy safeguards, access control, and domain expert oversight when applying such analyses to sensitive data.

REPRODUCIBILITY STATEMENT

We provide derivations and proofs in Appendix B and detailed experimental protocols in Appendix C, including dataset and model settings, removal-group construction, query selection, curvature construction, solver tolerances, clustering procedures, E2 coefficient solves, PBRF reference computation, timing protocols, and statistical testing. All main comparisons use fixed checkpoints, shared removal groups and queries, common curvature operators, and the same actual PBRF reference points within each setting.

REFERENCES

Miklós Máté Badó and Kristian Fenech. Clustered influence functions. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=Efjuj6kanc.

Juhan Bae, Nathan Hoyen Ng, Alston Lo, Marzyeh Ghassemi, and Roger Baker Grosse. If influence functions are the answer, then what is the question? In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=hzbguA9zMJ.

Arindam Banerjee, Srujana Merugu, Inderjit S Dhillon, and Joydeep Ghosh. Clustering with bregman divergences. *Journal of machine learning research*, 6(Oct):1705–1749, 2005.

Samyadeep Basu, Xuchen You, and Soheil Feizi. On second-order group influence functions for black-box predictions. In *International Conference on Machine Learning*, pp. 715–724. PMLR, 2020.

Samyadeep Basu, Phil Pope, and Soheil Feizi. Influence functions in deep learning are fragile. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=xHKVVHGDOEk.

Aleksandar Botev, Hippolyt Ritter, and David Barber. Practical Gauss-Newton optimisation for deep learning. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 557–565. PMLR, 06–11 Aug 2017. URL https://proceedings.mlr.press/v70/botev17a.html.

L. M. Bregman. The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming. *USSR Computational Mathematics and*

Mathematical Physics, 7(3):200–217, 1967. doi: 10.1016/0041-5553(67)90040-7. Translated from Zh. Vychisl. Mat. Mat. Fiz., 7(3):620–631, 1967.

Guillaume Charpiat, Nicolas Girard, Loris Felardos, and Yuliya Tarabalka. Input similarity from the neural network perspective. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 32*, pp. 5343–5352. Curran Associates, Inc., 2019. URL http://papers.nips.cc/paper/8775-input-similarity-from-the-neural-network-perspective.pdf.

R. Dennis Cook. Detection of influential observation in linear regression. *Technometrics*, 19(1): 15–18, February 1977. ISSN 0040-1706. doi: 10.1080/00401706.1977.10489493.

R Dennis Cook and Sanford Weisberg. Residuals and influence in regression. 1982.

Jiale Deng, Yanyan Shen, Xiaogang Shi, and Junjun Chai. Demix: Debugging training data with mixed data error types by investigating influence vectors. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2*, KDD '26, pp. 815–826, New York, NY, USA, 2026. Association for Computing Machinery. ISBN 9798400722592. doi: 10.1145/3770855.3817774. URL https://doi.org/10.1145/3770855.3817774.

Nikita Dhawan, Sicong Huang, Juhan Bae, and Roger Baker Grosse. Efficient parametric approximations of neural network function space distance. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 7795–7812. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/dhawan23a.html.

Jillian Fisher, Lang Liu, Krishna Pillutla, Yejin Choi, and Zaid Harchaoui. Influence diagnostics under self-concordance. In *International Conference on Artificial Intelligence and Statistics*, pp. 10028–10076. PMLR, 2023.

Amirata Ghorbani and James Zou. Data shapley: Equitable valuation of data for machine learning. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2242–2251. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/ghorbani19c.html.

Ryan Giordano, William Stephenson, Runjing Liu, Michael Jordan, and Tamara Broderick. A swiss army infinitesimal jackknife. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 1139–1147. PMLR, 2019.

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, 2023. URL https://arxiv.org/abs/2308.03296.

Zayd Hammoudeh and Daniel Lowd. Training data influence analysis and estimation: a survey. *Mach. Learn.*, 113(5):2351–2403, March 2024. ISSN 0885-6125. doi: 10.1007/s10994-023-06495-7. URL https://doi.org/10.1007/s10994-023-06495-7.

Frank R Hampel. The influence curve and its role in robust estimation. *Journal of the american statistical association*, 69(346):383–393, 1974.

Kazuaki Hanawa, Sho Yokoi, Satoshi Hara, and Kentaro Inui. Evaluation of similarity-based explanations, 2021. URL https://arxiv.org/abs/2006.04528.

Jaeseung Heo, Kyeongheung Yun, Seokwon Yoon, MoonJeong Park, Jungseul Ok, and Dongwoo Kim. Influence functions for edge edits in non-convex graph neural networks. In *Advances in Neural Information Processing Systems*, volume 38, pp. 166208–166234, 2025.

Jaeseung Heo, Kyeongheung Yun, Youngbin Choi, Sehyun Hwang, Jungseul Ok, and Dongwoo Kim. Interaction-aware influence functions for group attribution, 2026. URL https://arxiv.org/abs/2605.15675.

Sture Holm. A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6(2):65–70, 1979. ISSN 03036898, 14679469. URL http://www.jstor.org/stable/4615733.

Andrew Ilyas, Sung Min Park, Logan Engstrom, Guillaume Leclerc, and Aleksander Madry. Datamodels: Understanding predictions with data and data with predictions. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 9525–9587. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/ilyas22a.html.

Louis A. Jaeckel. The infinitesimal jackknife. (MM 72-1215-11), June 1972. URL https://faculty.washington.edu/fscholz/Reports/InfinitesimalJackknife.pdf. Unpublished memorandum; retypeset version by F. W. Scholz.

Ruoxi Jia, David Dao, Boxin Wang, Frances Ann Hubis, Nick Hynes, Nezihe Merve Gürel, Bo Li, Ce Zhang, Dawn Song, and Costas J. Spanos. Towards efficient data valuation based on the shapley value. In Kamalika Chaudhuri and Masashi Sugiyama (eds.), *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, volume 89 of *Proceedings of Machine Learning Research*, pp. 1167–1176. PMLR, 16–18 Apr 2019. URL https://proceedings.mlr.press/v89/jia19a.html.

Yegor Klochkov and Yang Liu. Revisiting inverse hessian vector products for calculating influence functions, 2024. URL https://arxiv.org/abs/2409.17357.

Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In *International conference on machine learning*, pp. 1885–1894. Pmlr, 2017.

Pang Wei W Koh, Kai-Siang Ang, Hubert Teo, and Percy S Liang. On the accuracy of influence functions for measuring group effects. *Advances in neural information processing systems*, 32, 2019.

Dan Ley, Suraj Srinivas, Shichang Zhang, Gili Rusak, and Himabindu Lakkaraju. Generalized group data attribution, 2024. URL https://arxiv.org/abs/2410.09940.

Stuart Lloyd. Least squares quantization in pcm. *IEEE transactions on information theory*, 28(2): 129–137, 1982.

James Martens. Deep learning via hessian-free optimization. In Johannes Fürnkranz and Thorsten Joachims (eds.), *Proceedings of the 27th International Conference on Machine Learning (ICML-10), June 21-24, 2010, Haifa, Israel*, pp. 735–742. Omnipress, 2010. URL https://icml.cc/Conferences/2010/papers/458.pdf.

Frank Nielsen and Richard Nock. Bregman sided and symmetrized centroids. In *ICPR 2008 19th International Conference on Pattern Recognition*, Los Alamitos, CA, USA, December 2008. IEEE Computer Society. doi: 10.1109/ICPR.2008.4761794. URL https://doi.ieeecomputersociety.org/10.1109/ICPR.2008.4761794.

Sung Min Park, Kristian Georgiev, Andrew Ilyas, Guillaume Leclerc, and Aleksander Madry. Trak: Attributing model behavior at scale. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org, 2023.

Daryl Pregibon. Logistic Regression Diagnostics. *The Annals of Statistics*, 9(4):705 – 724, 1981. doi: 10.1214/aos/1176345513. URL https://doi.org/10.1214/aos/1176345513.

Garima Pruthi, Frederick Liu, Satyen Kale, and Mukund Sundararajan. Estimating training data influence by tracing gradient descent. *Advances in neural information processing systems*, 33: 19920–19930, 2020.

Kamiar Rahnama Rad and Arian Maleki. A scalable estimate of the extra-sample prediction error via approximate leave-one-out, 2020. URL https://arxiv.org/abs/1801.10243.

Ittai Rubinstein and Samuel B. Hopkins. Rescaled influence functions: Accurate data attribution in high dimension. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=0T9lN3m3pc.

Ittai Rubinstein and Samuel B. Hopkins. On the accuracy of newton step and influence function data attributions, 2026. URL https://arxiv.org/abs/2512.12572.

Andrea Schioppa, Katja Filippova, Ivan Titov, and Polina Zablotskaia. Theoretical and practical perspectives on what influence functions do. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=gGl0n7Onug.

Nicol N. Schraudolph. Fast curvature matrix-vector products for second-order gradient descent. *Neural Computation*, 14(7):1723–1738, 2002. doi: 10.1162/08997660260028683.

Fulton Wang, Julius Adebayo, Sarah Tan, Diego Garcia-Olano, and Narine Kokhlikyan. Error discovery by clustering influence embeddings. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA, 2023. Curran Associates Inc.

Frank Wilcoxon. Individual comparisons by ranking methods. *Biometrics Bulletin*, 1(6):80–83, 1945. ISSN 00994987. URL http://www.jstor.org/stable/3001968.

Chih-Kuan Yeh, Joon Kim, Ian En-Hsu Yen, and Pradeep K Ravikumar. Representer point selection for explaining deep neural networks. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/8a7129b8f3edd95b7d969dfc2c8e9d9d-Paper.pdf.

A Related work

Training-data attribution. Training-data attribution explains model outputs in terms of the examples used for training; see Hammoudeh & Lowd (2024) for a survey. Representative instance-based approaches include representer points, gradient-similarity methods, TracIn, and influence functions (Koh & Liang, 2017; Yeh et al., 2018; Charpiat et al., 2019; Pruthi et al., 2020; Hanawa et al., 2021). Counterfactual attribution and valuation approaches include Data Shapley, Datamodels, and TRAK (Ghorbani & Zou, 2019; Jia et al., 2019; Ilyas et al., 2022; Park et al., 2023).

The PBRF as the reference response. Prior work has documented substantial gaps between neural-network influence estimates and literal retraining, as well as sensitivity to practical approximation choices (Basu et al., 2021; Schioppa et al., 2023). Bae et al. (2022) show that practical influence estimates for neural networks approximate the proximal Bregman response function (PBRF) more closely than leave-one-out retraining. The PBRF provides a nonlinear reference whose local approximation recovers the damped-GGN influence estimate. Following Bae et al. (2022), several studies have used the nonlinear PBRF response to evaluate influence approximations, including function-space approximations (Dhawan et al., 2023), language-model influence estimates (Grosse et al., 2023), inverse-curvature solvers (Klochkov & Liu, 2024). The PBRF has also been extended to graph edge edits (Heo et al., 2025).

Correcting additive group influence. The accuracy of influence, approximate leave-one/-k-out, and Newton-step estimators has also been studied under convexity and smoothness assumptions (Giordano et al., 2019; Rad & Maleki, 2020; Fisher et al., 2023; Rubinstein & Hopkins, 2026). Koh et al. (2019) find that high rank correlations between additive group influence and retraining effects can coexist with large errors in the predicted magnitudes. Their analysis uses a one-step Newton approximation that accounts for removal-induced curvature changes (Pregibon, 1981). Basu et al. (2020) derive second-order approximations of the parameter response under group reweighting, whereas Heo et al. (2026) retain the first-order parameter change and expand the target function to second order, yielding pairwise interaction terms. Rescaled Influence Functions (RIF) (Rubinstein & Hopkins, 2025) instead sum single-example Newton removal steps, preserving additivity of the estimated parameter change. Recent theory gives finite-sample analyses of IF and the single Newton step (NS) for convex ERM and identifies regimes in which NS is substantially more accurate than IF (Rubinstein & Hopkins, 2026).

Our decomposition initially preserves standard GI. E2 then optimizes a GGN-based quadratic surrogate of the PBRF edit objective within the span of the component contributions, and can therefore be viewed as a reduced-space Newton-type correction. The analysis of Rubinstein & Hopkins (2026) concerns a full-space Newton step for leave-group-out ERM, whereas E2 targets the PBRF and adjusts only the coefficients of component directions derived from the Bregman decomposition. E2 also keeps the component directions and target linearization fixed, rather than expanding the parameter response as in Basu et al. (2020) or the target as in Heo et al. (2026). RIF computes single-example Newton corrections independently; E2 instead fits the component coefficients jointly for the entire removal group.

Bregman centroids and information. Banerjee et al. (2005) develop centroid-based clustering with Bregman divergences and characterize Bregman information as the minimum average divergence to a single representative. We apply these established concepts to the scaled member-wise parameter responses induced by the linearized group PBRF. Our objective identity connects the additive estimate to its centroid representation, the dispersion it omits, and a multi-component decomposition. In the quadratic setting, target-projected BI is a scaled variance of individual influence scores; its association with nonlinear PBRF error is an empirical finding, not a consequence of the centroid identity.

Grouping and clustering for efficient attribution. Generalized Group Data Attribution (Ley et al., 2024) attributes model properties to groups rather than individual training examples, trading attribution granularity for computational efficiency. Its framework includes grouped versions of influence functions, TracIn (Pruthi et al., 2020), and TRAK (Park et al., 2023). Clustered Influence Functions (Badó & Fenech, 2026) cluster training-gradient sketches, cache inverse-curvature

responses for full-space cluster means, and answer subset queries by membership-weighted recombination. Their error analysis separates clustering scatter from solver error when approximating a fixed linearized influence estimate. We instead partition each specified removal group, obtain component contributions that recover its original GI estimate, and then optimize their combination to improve agreement with the nonlinear PBRF. Our Euclidean and Intra variants simplify partition construction and edit-curvature computation within this group-specific correction, rather than constructing a global response cache.

Influence representations for structure discovery. InfEmbed (Wang et al., 2023) applies k-means to influence embeddings of test examples to discover underperforming slices with similar training-data explanations. DeMix (Deng et al., 2026) represents each training example by its influences across validation predictions and uses multi-label classification to identify erroneous samples and their error types. Our construction instead starts from the influence objective itself: we prove that the linearized group PBRF objective equals a Bregman centroid objective, up to a parameter-independent constant (Theorem 3.1). This identity grounds the response points and their clustering geometry in the objective underlying standard GI. The multi-component extension refines this representation in the same geometry, with size-weighted contributions that exactly recover the additive estimate (Section 3.3). These components support both interpretation of the aggregate and its subsequent correction.

B Additional Theory and Proofs

We first give the local derivations used in Section 2, then prove the centroid and dispersion identities in Section 3 and the contribution decomposition in Section 3.3. The final subsection records the additional conditions needed for a nonquadratic Bregman generator.

B.1 Classical group influence by implicit differentiation

We give the group version of the classical derivation in Koh & Liang (2017), with the fixed-target additivity discussed by Koh et al. (2019). Assume that $\mathcal{J}$ and the losses L_i, $i \in \mathcal{S}$, are twice continuously differentiable on a neighborhood of θ_0, with

$$\nabla_\theta \mathcal{J}(\theta_0) = 0, \qquad H = \nabla^2_\theta \mathcal{J}(\theta_0) \succ 0.$$

These are local assumptions at an interior minimizer, not properties required of every trained original model. Define the stationarity map

$$F_{\mathcal{S}}(\theta, \epsilon) := \nabla_\theta \mathcal{J}(\theta) + \epsilon \sum_{i \in \mathcal{S}} \nabla_\theta L_i(\theta).$$

Since $F_{\mathcal{S}}(\theta_0, 0) = 0$ and $D_\theta F_{\mathcal{S}}(\theta_0, 0) = H$, the implicit-function theorem gives a unique continuously differentiable stationary branch $\theta^{\mathrm{loc}}_{\mathcal{S}}(\epsilon)$ near $(\theta_0, 0)$. Continuity of the Hessian makes this a branch of strict local minimizers for sufficiently small $|\epsilon|$.

Applying the chain rule to $F_{\mathcal{S}}(\theta^{\mathrm{loc}}_{\mathcal{S}}(\epsilon), \epsilon) = 0$ gives

$$D_\theta F_{\mathcal{S}}(\theta^{\mathrm{loc}}_{\mathcal{S}}(\epsilon), \epsilon) \frac{d\theta^{\mathrm{loc}}_{\mathcal{S}}(\epsilon)}{d\epsilon} + \partial_\epsilon F_{\mathcal{S}}(\theta^{\mathrm{loc}}_{\mathcal{S}}(\epsilon), \epsilon) = 0. \tag{21}$$

At $\epsilon = 0$, the two derivatives of $F_{\mathcal{S}}$ are H and $c_{\mathcal{S}} = \sum_{i \in \mathcal{S}} z_i$, respectively, so

$$\left.\frac{d\theta^{\mathrm{loc}}_{\mathcal{S}}(\epsilon)}{d\epsilon}\right|_{\epsilon=0} = -H^{-1} c_{\mathcal{S}}.$$

Consequently, for fixed $\mathcal{S}$,

$$\theta^{\mathrm{loc}}_{\mathcal{S}}(\epsilon) = \theta_0 - \epsilon H^{-1} c_{\mathcal{S}} + o(|\epsilon|), \qquad \epsilon \to 0.$$

If the perturbed objectives are globally strongly convex and attain their minima for sufficiently small $|\epsilon|$, this branch is their unique global minimizer $\theta^\star_{\mathcal{S}}(\epsilon)$. Otherwise, the conclusion is local.

For a scalar target f differentiable at θ_0, a second application of the chain rule yields

$$\left.\frac{d}{d\epsilon} f(\theta^{\mathrm{loc}}_{\mathcal{S}}(\epsilon))\right|_{\epsilon=0} = \nabla_\theta f(\theta_0)^\top \left.\frac{d\theta^{\mathrm{loc}}_{\mathcal{S}}(\epsilon)}{d\epsilon}\right|_{\epsilon=0} = -\nabla_\theta f(\theta_0)^\top H^{-1} c_{\mathcal{S}}.$$

Thus

$$f(\theta^{\mathrm{loc}}_{\mathcal{S}}(\epsilon)) - f(\theta_0) = -\epsilon \nabla_\theta f(\theta_0)^\top H^{-1} c_{\mathcal{S}} + o(|\epsilon|).$$

The first-order term is additive over disjoint groups for a fixed original model and target. Setting $\epsilon = -1/N$ in the parameter expansion recovers the removal estimate in Equation (1); the local derivation alone does not guarantee accuracy at that finite weight.

B.2 PBRF influence by the chain rule and its quadratic objective

We derive the first-order response of Equation (4) by differentiating its stationarity condition, then identify the quadratic objective in Equation (5). This is the group version of the PBRF construction of Bae et al. (2022), with the negative removal weight used throughout this paper. Assume that $h_\theta(x_v)$ is twice continuously differentiable near θ_0, and that $\ell(\cdot, y_v)$ is twice continuously differentiable near $h_{\theta_0}(x_v)$ and convex on a convex domain containing the attainable outputs. Write $h_v(\theta) = h_\theta(x_v) \in \mathbb{R}^C$, where C is the output dimension, and set

$$h^s_v = h_v(\theta_0), \qquad J_v(\theta) = D_\theta h_v(\theta) \in \mathbb{R}^{C \times d}, \qquad H_v(\theta) = \nabla^2_u \ell(h_v(\theta), y_v) \in \mathbb{R}^{C \times C}.$$

Thus $J_v = J_v(\theta_0)$ and $H_v = H_v(\theta_0)$ agree with the notation in Section 2.1.

Source derivatives of the output-Bregman term. Define

$$B(\theta) := \frac{1}{N}\sum_{v=1}^{N} b_v(\theta), \qquad b_v(\theta) := D_{\ell(\cdot,y_v)}(h_v(\theta), h_v^s),$$

and let $e_v(\theta) = \nabla_u \ell(h_v(\theta), y_v) - \nabla_u \ell(h_v^s, y_v)$. The chain and product rules give the exact identities

$$\nabla_\theta b_v(\theta) = J_v(\theta)^\top e_v(\theta),$$
$$\nabla_\theta^2 b_v(\theta) = J_v(\theta)^\top H_v(\theta) J_v(\theta) + \sum_{a=1}^{C} e_{v,a}(\theta)\, \nabla_\theta^2 h_{v,a}(\theta).$$

At θ_0, $e_v(\theta_0) = 0$, so the terms involving second derivatives of the model vanish exactly. Therefore

$$B(\theta_0) = 0, \qquad \nabla_\theta B(\theta_0) = 0, \qquad \nabla_\theta^2 B(\theta_0) = G_0 := \frac{1}{N}\sum_{v=1}^{N} J_v^\top H_v J_v.$$

In particular, G_0 is the exact Hessian of the output-Bregman term at θ_0, not an approximation obtained by discarding nonzero model-Hessian terms there. It need not equal the Hessian of the original training loss. Since $H_v \succeq 0$ and $\lambda > 0$, $G = G_0 + \lambda I_d \succ 0$.

Differentiating the minimizing branch. At $\epsilon = 0$, nonnegativity of the output divergences and the positive proximal penalty make θ_0 the unique global PBRF minimizer. This requires no condition $\nabla_\theta \mathcal{J}(\theta_0) = 0$. Let

$$F_{\mathcal{S}}^{\mathrm{PBRF}}(\theta, \epsilon) := \nabla_\theta B(\theta) + \lambda(\theta - \theta_0) + \epsilon \sum_{i \in \mathcal{S}} \nabla_\theta L_i(\theta).$$

Its derivatives at the unperturbed point are

$$D_\theta F_{\mathcal{S}}^{\mathrm{PBRF}}(\theta_0, 0) = G, \qquad \partial_\epsilon F_{\mathcal{S}}^{\mathrm{PBRF}}(\theta_0, 0) = c_{\mathcal{S}}.$$

The implicit-function theorem gives a continuously differentiable branch of strict local minimizers $\theta_{\mathcal{S}}^{\mathrm{PBRF,loc}}(\epsilon)$ through θ_0. Differentiating $F_{\mathcal{S}}^{\mathrm{PBRF}}(\theta_{\mathcal{S}}^{\mathrm{PBRF,loc}}(\epsilon), \epsilon) = 0$ by the chain rule, as in Equation (21), gives

$$G\, \frac{d\theta_{\mathcal{S}}^{\mathrm{PBRF,loc}}(\epsilon)}{d\epsilon}\bigg|_{\epsilon=0} + c_{\mathcal{S}} = 0, \qquad \frac{d\theta_{\mathcal{S}}^{\mathrm{PBRF,loc}}(\epsilon)}{d\epsilon}\bigg|_{\epsilon=0} = -G^{-1} c_{\mathcal{S}}.$$

The group-Hessian term $\epsilon \sum_{i \in \mathcal{S}} \nabla_\theta^2 L_i(\theta)$ vanishes at $\epsilon = 0$; no assumption that it is identically zero is needed. For fixed $\mathcal{S}$,

$$\theta_{\mathcal{S}}^{\mathrm{PBRF,loc}}(\epsilon) = \theta_0 - \epsilon G^{-1} c_{\mathcal{S}} + o(|\epsilon|), \qquad \epsilon \to 0. \tag{22}$$

For any scalar target f differentiable at θ_0, the chain rule then gives

$$\begin{aligned}\frac{d}{d\epsilon} f(\theta_{\mathcal{S}}^{\mathrm{PBRF,loc}}(\epsilon))\bigg|_{\epsilon=0} &= \nabla_\theta f(\theta_0)^\top \left.\frac{d\theta_{\mathcal{S}}^{\mathrm{PBRF,loc}}(\epsilon)}{d\epsilon}\right|_{\epsilon=0} \\ &= -\nabla_\theta f(\theta_0)^\top G^{-1} c_{\mathcal{S}} = -q^\top c_{\mathcal{S}}, \quad q = G^{-1}\nabla_\theta f(\theta_0).\end{aligned}$$

Multiplying by ϵ gives $\widehat{A}_{\mathcal{S}}(\epsilon)$ in Equation (3). This does not identify the first-order estimate with the nonlinear target change at a finite removal weight.

The quadratic objective for the first-order response. The affine approximation in Equation (22) is the unique global minimizer of

$$Q_{\mathcal{S}}^{\mathrm{lin}}(\theta; \epsilon) = \tfrac{1}{2}(\theta - \theta_0)^\top G(\theta - \theta_0) + \epsilon c_{\mathcal{S}}^\top(\theta - \theta_0),$$

because its stationarity equation is $G(\theta - \theta_0) + \epsilon c_{\mathcal{S}} = 0$. Thus implicit differentiation and the quadratic objective in Equation (5) give the same damped-GGN estimate.

The objective approximation can also be checked directly. For $\delta = \theta - \theta_0$,

$$B(\theta_0 + \delta) = \tfrac{1}{2}\delta^\top G_0 \delta + o(\|\delta\|_2^2),$$
$$\sum_{i\in\mathcal{S}} L_i(\theta_0 + \delta) = \sum_{i\in\mathcal{S}} L_i(\theta_0) + c_{\mathcal{S}}^\top \delta + O(\|\delta\|_2^2).$$

Hence, as $\delta \to 0$ with $\mathcal{S}$ fixed,

$$Q_{\mathcal{S}}^{\mathrm{PBRF}}(\theta_0 + \delta; \epsilon) = \epsilon \sum_{i\in\mathcal{S}} L_i(\theta_0) + Q_{\mathcal{S}}^{\mathrm{lin}}(\theta_0 + \delta; \epsilon) + R_{\mathcal{S}}(\delta, \epsilon),$$
$$R_{\mathcal{S}}(\delta, \epsilon) = o(\|\delta\|_2^2) + O(|\epsilon|\|\delta\|_2^2).$$

This retains the unperturbed Bregman term to second order and the group perturbation to first order in δ. It is not the full second-order Taylor objective at a fixed nonzero ϵ, which would also contain the group-Hessian term. The centroid characterization in Section 3 is exact for $Q_{\mathcal{S}}^{\mathrm{lin}}$; the local argument does not guarantee a unique global nonlinear PBRF minimizer at finite removal weights. For piecewise-smooth networks, this differentiation argument applies on neighborhoods where the required derivatives exist. The algebraic centroid identity itself only requires the specified gradients and $G \succ 0$.

B.3 Bregman centroids and proof of the main characterization

We first state the centroid identity underlying the definition of Bregman information (Banerjee et al., 2005, Proposition 1).

Lemma B.1 (Fixed-group Bregman centroid identity)**.** *Let Ψ be differentiable and convex on an open convex set Ω, and let $a_1, \dots, a_m \in \Omega$, with $m \geq 1$ and $\bar{a} = m^{-1}\sum_i a_i$. For every $\mu \in \Omega$,*

$$\frac{1}{m}\sum_{i=1}^{m} D_\Psi(a_i, \mu) = \frac{1}{m}\sum_{i=1}^{m} D_\Psi(a_i, \bar{a}) + D_\Psi(\bar{a}, \mu). \tag{23}$$

Hence $\bar{a}$ is a minimizer. It is unique if Ψ is strictly convex.

Proof. Since $\sum_i (a_i - \bar{a}) = 0$, the difference between the first two terms in Equation (23) is

$$\Psi(\bar{a}) - \Psi(\mu) - \nabla\Psi(\mu)^\top(\bar{a} - \mu) = D_\Psi(\bar{a}, \mu).$$

Convexity makes this quantity nonnegative; strict convexity makes it zero only when $\mu = \bar{a}$. □

Proof of Theorem 3.1. Let $\delta = \theta - \theta_0$ and use the response points from Equation (7). Completing the square gives

$$\begin{aligned} D_{\Phi_G}\left(\theta, \widetilde{\theta}_i^{(m)}(\epsilon)\right) &= \tfrac{1}{2}(\delta + \epsilon m G^{-1} z_i)^\top G (\delta + \epsilon m G^{-1} z_i) \\ &= \tfrac{1}{2}\delta^\top G \delta + \epsilon m z_i^\top \delta + \tfrac{\epsilon^2 m^2}{2} z_i^\top G^{-1} z_i. \end{aligned}$$

Averaging over $\mathcal{S}$ yields

$$\frac{1}{m}\sum_{i\in\mathcal{S}} D_{\Phi_G}\left(\theta, \widetilde{\theta}_i^{(m)}(\epsilon)\right) = Q_{\mathcal{S}}^{\mathrm{lin}}(\theta; \epsilon) + \frac{\epsilon^2 m}{2}\sum_{i\in\mathcal{S}} z_i^\top G^{-1} z_i.$$

Rearranging and using the symmetry of the quadratic divergence gives Equation (8) with the stated constant $K_{\mathcal{S}}(\epsilon)$. By Lemma B.1, its unique minimizer is

$$\frac{1}{m}\sum_{i\in\mathcal{S}} \widetilde{\theta}_i^{(m)}(\epsilon) = \theta_0 - \epsilon G^{-1}\sum_{i\in\mathcal{S}} z_i = \widehat{\theta}_{\mathcal{S}}(\epsilon).$$

The identity holds for every θ and every real ϵ, including $\epsilon = 0$. □

B.4 Full and target-projected Bregman information

Set $\bar{z}_{\mathcal{S}} = m^{-1}\sum_{i\in\mathcal{S}} z_i$. The centered response points satisfy

$$\widetilde{\theta}_i^{(m)}(\epsilon) - \widehat{\theta}_{\mathcal{S}}(\epsilon) = -\epsilon m G^{-1}(z_i - \bar{z}_{\mathcal{S}}).$$

Substitution into the quadratic divergence proves

$$\begin{aligned}\frac{1}{m}\sum_{i\in\mathcal{S}} D_{\Phi_G}\left(\widetilde{\theta}_i^{(m)}(\epsilon), \widehat{\theta}_{\mathcal{S}}(\epsilon)\right) &= \frac{\epsilon^2 m^2}{2m}\sum_{i\in\mathcal{S}}(z_i - \bar{z}_{\mathcal{S}})^\top G^{-1}(z_i - \bar{z}_{\mathcal{S}}) \\ &= \epsilon^2 m^2 \mathcal{B}_G(\mathcal{S}).\end{aligned}$$

Thus the normalized full BI in Equation (10) removes only the common perturbation and group-size scale. It still depends on the fixed source and curvature matrix G.

For a differentiable scalar target f, write $g = \nabla f(\theta_0)$ and $q = G^{-1}g$. Define the linearized target value at each response point by

$$t_i^{\mathrm{lin}} := g^\top\left(\widetilde{\theta}_i^{(m)}(\epsilon) - \theta_0\right) = -\epsilon m q^\top z_i.$$

Its arithmetic mean is $\widehat{A}_{\mathcal{S}}(\epsilon)$. For $\psi(t) = t^2/2$,

$$\frac{1}{m}\sum_{i\in\mathcal{S}} D_\psi(t_i^{\mathrm{lin}}, \widehat{A}_{\mathcal{S}}(\epsilon)) = \epsilon^2 m^2 \mathcal{B}_{\mathrm{proj}}(\mathcal{S}).$$

The singleton contribution uses weight ϵ, rather than $m\epsilon$, and equals $\widehat{A}_{\{i\}}(\epsilon) = -\epsilon q^\top z_i$. Consequently,

$$\tfrac{1}{2}\operatorname{Var}_{i\sim\mathrm{Unif}(\mathcal{S})}\left(\widehat{A}_{\{i\}}(\epsilon)\right) = \epsilon^2 \mathcal{B}_{\mathrm{proj}}(\mathcal{S}).$$

These are exact identities for linearized target values, not for nonlinear evaluations $f(\widetilde{\theta}_i^{(m)}(\epsilon)) - f(\theta_0)$. The variance uses denominator m, not $m-1$.

B.5 Component centroids, conservation, and whitening

We first show that component centroids preserve the original group estimate for any partition, as used in Equation (15). We then express the within-component Bregman objective in whitened coordinates. Throughout, $G \succ 0$, $m = |\mathcal{S}| \geq 1$, and $\mathcal{C}$ partitions $\mathcal{S}$ into k nonempty components, with $1 \leq k \leq m$.

Conservation of the group estimate. For $C \in \mathcal{C}$, define $\pi_C = |C|/m$, $\bar{z}_C = |C|^{-1}\sum_{i\in C} z_i$, and

$$\widetilde{\theta}_C^{(m)}(\epsilon) := \frac{1}{|C|}\sum_{i\in C}\widetilde{\theta}_i^{(m)}(\epsilon) = \theta_0 - \epsilon m G^{-1}\bar{z}_C. \tag{24}$$

For a fixed target, let $g = \nabla_\theta f(\theta_0)$ and $q = G^{-1}g$. The component contribution is

$$\widehat{A}_C(\epsilon) = \pi_C g^\top\left(\widetilde{\theta}_C^{(m)}(\epsilon) - \theta_0\right) = -\epsilon q^\top \sum_{i\in C} z_i.$$

Since the components partition $\mathcal{S}$, summing their gradient sums gives $c_{\mathcal{S}}$, and therefore

$$\sum_{C\in\mathcal{C}}\widehat{A}_C(\epsilon) = \widehat{A}_{\mathcal{S}}(\epsilon), \qquad \sum_{C\in\mathcal{C}}\pi_C\widetilde{\theta}_C^{(m)}(\epsilon) = \widehat{\theta}_{\mathcal{S}}(\epsilon).$$

Thus contributions add, whereas centroids recover the group centroid by a size-weighted average. Both identities hold for any partition; they do not depend on optimizing an assignment objective. The centroid of a component in the original m-scaled cloud is not generally the standalone linearized PBRF estimate for that subgroup. Their displacements are related by

$$\widehat{\theta}_C(\epsilon) - \theta_0 = \pi_C\left(\widetilde{\theta}_C^{(m)}(\epsilon) - \theta_0\right).$$

These identities concern the fixed additive estimate and do not imply additivity of nonlinear PBRF changes.

Choosing components in the same geometry. For a prescribed k, the partition criterion in Equation (13) is

$$\min_{\substack{\mathcal{C} \text{ partitions } \mathcal{S} \\ |\mathcal{C}|=k}} W_{\mathcal{C}}, \qquad W_{\mathcal{C}} := \frac{1}{m} \sum_{C \in \mathcal{C}} \sum_{i \in C} D_{\Phi_G}\left(\widetilde{\theta}_i^{(m)}(\epsilon), \widetilde{\theta}_C^{(m)}(\epsilon)\right).$$

For each fixed partition, Lemma B.1 shows that its component means minimize the divergence over representatives. Selecting components by this criterion refines the description of the cloud without changing the original group estimate. It does not guarantee that the resulting partition maximizes cancellation visible to any particular target.

The quadratic generator and its conjugate are

$$\Phi_G(\theta) = \tfrac{1}{2}\theta^\top G\theta, \qquad \Phi_G^*(z) = \tfrac{1}{2} z^\top G^{-1} z.$$

Proposition B.2 (Whitened response-space equivalence)**.** *Define*

$$w_i := G^{-1/2} z_i, \qquad \bar{w}_C := \frac{1}{|C|} \sum_{i \in C} w_i,$$

where $G^{-1/2}$ is the symmetric positive definite inverse square root. For every partition $\mathcal{C}$ into nonempty components,

$$W_{\mathcal{C}} = \frac{\epsilon^2 m^2}{2m} \sum_{C \in \mathcal{C}} \sum_{i \in C} \|w_i - \bar{w}_C\|_2^2.$$

For $\epsilon \neq 0$, minimizing $W_{\mathcal{C}}$ and the unscaled within-component Euclidean sum of squares gives the same globally optimal partitions.

Proof. From Equations (7) and (24),

$$\widetilde{\theta}_i^{(m)}(\epsilon) - \widetilde{\theta}_C^{(m)}(\epsilon) = -\epsilon m G^{-1}(z_i - \bar{z}_C).$$

Substitution into the quadratic divergence gives

$$D_{\Phi_G}\left(\widetilde{\theta}_i^{(m)}(\epsilon), \widetilde{\theta}_C^{(m)}(\epsilon)\right) = \frac{\epsilon^2 m^2}{2} \|w_i - \bar{w}_C\|_2^2.$$

Summing proves the identity. For nonzero ϵ, the scale factor is positive and independent of the partition. For $\epsilon = 0$, all parameter-response points coincide and every partition minimizes $W_{\mathcal{C}}$, so equivalence with the unscaled Euclidean optimization need not hold. □

The equivalence is a change of coordinates, not a guarantee that a particular k-means initialization finds a global optimum.

Within- and between-component dispersion. Let $a_i = \widetilde{\theta}_i^{(m)}(\epsilon)$, $a_C = \widetilde{\theta}_C^{(m)}(\epsilon)$, and $\bar{a} = \widehat{\theta}_{\mathcal{S}}(\epsilon)$. Applying Lemma B.1 within each component and summing gives

$$\underbrace{\frac{1}{m} \sum_{i \in \mathcal{S}} D_{\Phi_G}(a_i, \bar{a})}_{T} = \underbrace{\frac{1}{m} \sum_{C \in \mathcal{C}} \sum_{i \in C} D_{\Phi_G}(a_i, a_C)}_{W_{\mathcal{C}}} + \underbrace{\sum_{C \in \mathcal{C}} \pi_C D_{\Phi_G}(a_C, \bar{a})}_{B_{\mathcal{C}}}.$$

All three dispersions share the same positive scaling under whitening when $\epsilon \neq 0$. This decomposes the cloud's dispersion, not its nonlinear approximation error.

B.6 Beyond quadratic generators: a dual-space centroid representation

The preceding results use a quadratic generator. For completeness, we state when a linear tilt of a general generator has a shifted-center representation. The gradient and conjugacy properties used here are standard Legendre duality; see Banerjee et al. (2005, Theorem 2 and Appendix A).

Let $\Phi : \mathbb{R}^d \to (-\infty, +\infty]$ be proper, closed, convex, and Legendre, with nonempty $\Theta = \operatorname{int}(\operatorname{dom}\Phi)$. Define its Fenchel conjugate $\Phi^*(\eta) = \sup_\theta\{\eta^\top\theta - \Phi(\theta)\}$ and $\Theta^* = \operatorname{int}(\operatorname{dom}\Phi^*)$. Then

$$\nabla\Phi : \Theta \longrightarrow \Theta^* \quad \text{is bijective, with inverse } \nabla\Phi^*.$$

Theorem B.3 (Center shift under a linear tilt)**.** *Fix $\theta_0 \in \Theta$, $a \in \mathbb{R}^d$, and $\eta = \nabla\Phi(\theta_0) - a$. There exist $\widetilde{\theta} \in \Theta$ and a constant K such that*

$$D_\Phi(\theta, \theta_0) + a^\top(\theta - \theta_0) = D_\Phi(\theta, \widetilde{\theta}) + K \quad \textit{for every } \theta \in \Theta \tag{25}$$

if and only if

$$\eta \in \Theta^* = \nabla\Phi(\Theta).$$

In that case the center and constant are unique:

$$\widetilde{\theta} = \nabla\Phi^*(\eta), \qquad K = -D_\Phi(\theta_0, \widetilde{\theta}). \tag{26}$$

Moreover,

$$D_\Phi(\theta, \widetilde{\theta}) = D_{\Phi^*}(\eta, \nabla\Phi(\theta)).$$

Proof. Differentiating Equation (25) in θ forces $\nabla\Phi(\widetilde{\theta}) = \nabla\Phi(\theta_0) - a = \eta$. This proves necessity and uniqueness of the center. Conversely, if $\eta \in \Theta^*$, the center in Equation (26) exists. Expanding the divergences shows that the difference in Equation (25) is constant; evaluating at $\theta = \theta_0$ determines K. Finally, $\Phi^*(\nabla\Phi(v)) = v^\top\nabla\Phi(v) - \Phi(v)$ gives $D_\Phi(u, v) = D_{\Phi^*}(\nabla\Phi(v), \nabla\Phi(u))$, proving the last identity. □

The range condition is substantive: $\Phi(t) = e^t$ is Legendre on $\mathbb{R}$, but its gradient range is $(0, \infty)$. For $\theta_0 = 0$ and $a = 2$, the required dual coordinate is -1, so no interior shifted center exists. For the positive definite quadratic generator, the gradient range is all of $\mathbb{R}^d$.

Theorem B.4 (Centroid representation in dual coordinates)**.** *With Φ as above, fix $\theta_0 \in \Theta$, a nonempty group $\mathcal{S}$ of size m, gradients $z_i \in \mathbb{R}^d$, and $\epsilon \in \mathbb{R}$. Suppose*

$$\eta_i^{(m)} := \nabla\Phi(\theta_0) - \epsilon m z_i \in \Theta^* \quad \textit{for every } i \in \mathcal{S}.$$

Let

$$\widetilde{\theta}_i^{(m)} := \nabla\Phi^*(\eta_i^{(m)}), \qquad \bar{\eta}_\mathcal{S} := \frac{1}{m}\sum_{i\in\mathcal{S}} \eta_i^{(m)} = \nabla\Phi(\theta_0) - \epsilon c_\mathcal{S}.$$

For every $\theta \in \Theta$,

$$\begin{aligned} D_\Phi(\theta, \theta_0) + \epsilon c_\mathcal{S}^\top(\theta - \theta_0) &= \frac{1}{m}\sum_{i\in\mathcal{S}} D_\Phi(\theta, \widetilde{\theta}_i^{(m)}) + K_\mathcal{S}^\Phi(\epsilon) \\ &= \frac{1}{m}\sum_{i\in\mathcal{S}} D_{\Phi^*}(\eta_i^{(m)}, \nabla\Phi(\theta)) + K_\mathcal{S}^\Phi(\epsilon), \end{aligned}$$

where

$$K_\mathcal{S}^\Phi(\epsilon) := -\frac{1}{m}\sum_{i\in\mathcal{S}} D_\Phi(\theta_0, \widetilde{\theta}_i^{(m)}).$$

The unique minimizer is

$$\widehat{\theta}_\mathcal{S}^\Phi(\epsilon) = \nabla\Phi^*(\bar{\eta}_\mathcal{S}).$$

Proof. Distribute the linear tilt as

$$D_\Phi(\theta, \theta_0) + \epsilon c_\mathcal{S}^\top(\theta - \theta_0) = \frac{1}{m}\sum_{i\in\mathcal{S}} \big[D_\Phi(\theta, \theta_0) + \epsilon m z_i^\top(\theta - \theta_0)\big].$$

Apply Theorem B.3 to each summand with $a = \epsilon m z_i$, then use its dual identity. This proves both representations and their common constant. Convexity of Θ^* implies $\bar{\eta}_\mathcal{S} \in \Theta^*$. By Lemma B.1, the unique optimal representative in the dual objective is $\nabla\Phi(\theta) = \bar{\eta}_\mathcal{S}$, giving the stated minimizer. □

For a nonquadratic generator, the arithmetic centroid is in dual coordinates, not generally in parameter space, and the divergence arguments cannot generally be reversed. The member-wise representation also requires more than existence of an aggregate shifted center: the latter needs only $\bar{\eta}_\mathcal{S} \in \Theta^*$, whereas the former needs every $\eta_i^{(m)} \in \Theta^*$. This is a statement about a Bregman divergence plus a linear tilt, not a claim that the full nonlinear PBRF has this form.

Quadratic specialization and scope. For $\Phi_G(\theta) = \frac{1}{2}\theta^\top G\theta$ with $G \succ 0$, $\Theta^* = \mathbb{R}^d$, $\nabla\Phi_G(\theta) = G\theta$, and $\nabla\Phi_G^*(\eta) = G^{-1}\eta$. The range condition is then automatic, the shifted points are $\theta_0 - \epsilon m G^{-1} z_i$, and the dual-centroid formula recovers Theorem 3.1. For a nonquadratic generator, averaging in dual coordinates does not in general commute with $\nabla\Phi^*$. Consequently, this extension does not establish the additive parameter-response decomposition in Equation (15) outside the quadratic setting, nor does it provide an accuracy guarantee for E2 or nonlinear PBRF.

C Experimental Details

This section specifies the common protocol, numerical methods, and cost model.

C.1 Common protocol

Each comparison uses a fixed original checkpoint, 100 random removal groups per removal ratio, and 100 fixed test queries. All compared methods use the same original model, removal groups, test queries, curvature operator, and actual PBRF reference points. Partitions and coefficients are fitted using training-side quantities only and reused across queries. Unless stated otherwise, a method predicts the loss change of query j by applying the fixed target gradient $\nabla_\theta f_j(\theta_0)$ to its estimated parameter change, as in Equation (3). The reference is $f_j(\theta_S^{\mathrm{PBRF}}) - f_j(\theta_0)$.

C.2 Datasets and original models

Table 3: Models and training-set sizes. The ridge is shared by G and the PBRF proximal term; it is distinct from training regularization.

Dataset	Model	Classes	N	Ridge
Dogfish	Logistic head, 2,048 features + intercept	2	1,800	0.01
MNIST	ReLU MLP, 784–512–256–10	10	60,000	0.1001
CIFAR-10	Myrtle/DAWNBench ResNet9	10	50,000	0.3005

Dogfish. The head uses fixed Inception-v3 features (Koh & Liang, 2017; Koh et al., 2019), an ℓ_2-regularized logistic initialization, and full-batch L-BFGS refinement. Its objective is mean binary cross-entropy plus $10^{-4}\|\theta\|^2$, including the intercept.

MNIST. We use Adam for 30 epochs (learning rate 10^{-3}, batch size 256), then full-batch L-BFGS (learning rate 0.5, history 20, at most 80 steps). The objective is mean cross-entropy plus $\frac{1}{2}10^{-4}\|\theta\|^2$.

ResNet9. We use Nesterov SGD for 24 epochs (batch size 512, momentum 0.9, weight decay 5×10^{-4}). The learning rate follows a piecewise-linear $0 \to 0.4 \to 0$ schedule, peaking at epoch 5. Training uses channel normalization, random crops with padding 4, horizontal flips, and 8×8 Cutout. Full-batch L-BFGS refinement uses clean inputs, history 10, and successive (learning rate, step budget) pairs $(0.1, 10)$, $(0.01, 20)$, and $(0.01, 60)$, without line search. Influence and PBRF evaluations use clean inputs; ResNet9 batch-normalization buffers remain fixed.

C.3 Groups, targets, and evaluation

Sampling and targets. Each group is sampled uniformly without replacement, without class quotas; independently drawn groups may overlap. Thus, 100 groups need not be disjoint subsets of the training set. Dogfish queries come from its 400-example evaluation pool. MNIST uses ten queries per class from a 6,000-example evaluation pool, disjoint from the 4,000 validation examples; ResNet9 also uses ten test queries per class. The class profiles in Appendix C.7 instead average over each class in the full 10,000-example MNIST test set.

Query-wise metrics. For each fixed query, we compare the method's predictions with the actual PBRF loss changes across the 100 groups, using

$$R^2 = 1 - \frac{\mathrm{SSE}}{\mathrm{SST}}.$$

Here SSE is the sum of squared prediction errors, and SST is the sum of squared deviations of the reference values from their mean across groups. No regression, affine recalibration, or rescaling is fitted to the predictions. Spearman correlation measures ranking across groups for the same query, whereas raw R^2 measures scale-normalized value accuracy and may be negative. We report medians

and, where indicated, interquartile ranges across queries. Measured results are displayed to three significant figures; a displayed value of 1.00 may reflect rounding rather than exact agreement. Tests and reported differences use unrounded scores, and reproducibility-critical settings retain their exact values. We neither average query losses before computing these metrics nor pool query–group pairs into one correlation.

Within each comparison, all methods and both metrics use a common query mask. A query is excluded if its reference or any compared prediction is numerically constant: the range across groups is at most 10^{-14} plus 10^{-10} times the largest absolute value. This also excludes constant predictions for which raw R^2 alone would be defined. The main Dogfish and MNIST comparisons retain all 100 queries; the ResNet9 four-method comparison retains 97. All retain 100 groups.

BI normalization and figure selection. The BI plots use $m\mathcal{B}_{\mathrm{proj}}(\mathcal{S})$ without the common removal factor N^{-2}. Full BI and Euclidean dispersion use the corresponding summed normalization, with Euclidean dispersion defined by $\frac{1}{2}\sum_{i\in\mathcal{S}}\|z_i-\bar{z}_{\mathcal{S}}\|^2$. These fixed-size positive scalings do not change Spearman correlation with squared GI error. Each BI panel uses the query nearest the median query-wise BI–error correlation for the corresponding model. Prediction-scatter annotations summarize median query-wise performance, not the scores for the displayed query. Appendix D.2 compares all three dispersion measures on Dogfish and ten-class MNIST at 1% removal. The three-model projected-BI size sweep is reported separately in Appendix D.1.

Paired significance tests. For each metric, we pair the Euclidean Joint or Intra E2 score at $k = 2$ with the single-component E2 ($k = 1$) score for the same query. We use two-sided Wilcoxon signed-rank tests (Wilcoxon, 1945) of the query-wise differences. The null is that the paired-difference distribution is symmetric about zero; this is not a test solely of the difference between the two marginal medians. We round differences to 12 decimal places before ranking to merge floating-point ties, discard zero differences from the rank statistic, assign average ranks to tied absolute differences, and use the tie-corrected normal approximation without continuity correction (SciPy 1.13.1). The displayed medians use the original unrounded scores and the common query mask above.

Holm step-down correction (Holm, 1979) is applied to one family of 26 tests: 13 available method–setting comparisons times two metrics, raw R^2 and Spearman. The comparisons are Joint at Dogfish 1% and Joint/Intra at Dogfish 10%/30%, MNIST 1%/10%/30%, and ResNet9 1%. All comparisons, including decreases, enter this family. Table 6 gives the medians and adjusted p-values. We describe a significant difference as an improvement only if both the difference of marginal medians and the median paired difference are positive. The Spearman tests compare the two methods' query-wise correlations; they do not test correlation against zero. These query-paired tests condition on the fixed checkpoint and sampled removal groups.

C.4 Curvature and component construction

We use the full-training-set GGN with the ridge in Table 3 and removal weight $\epsilon = -1/N$. Per-example gradients and edit curvature exclude training regularization, which is not added again to the PBRF objective. Dogfish uses dense FP64 algebra; MNIST uses matrix-free FP64 products, with checked relative residuals of 10^{-8} for GI and at most 5×10^{-8} for component directions. ResNet9 uses mixed-precision inverse solves with FP64 residual checks at relative tolerance 1.1×10^{-6}.

Bregman and Euclidean clustering apply the G^{-1} and identity metrics to raw gradients, respectively. Euclidean clustering does not replace G in the response directions or the common E2 term. Dogfish uses ten Lloyd restarts (Lloyd, 1982), a random first center followed by farthest-point initialization, and at most 300 iterations per restart; it stops on unchanged assignments or relative center movement at most 10^{-8} and selects the lowest-SSE restart. MNIST and ResNet9 use two restarts with balanced random initial assignments, at most 100 iterations each, and convergence on unchanged assignments. We require both restarts to converge and retain the partition with the lower clustering objective.

C.5 PBRF reference computation

Each group's actual PBRF reference point is computed once using Equation (4) and reused across methods, component counts, and queries. Table 4 specifies the initialization and optimization budget.

The MNIST budget consists of up to 200 optimizer iterations, not necessarily 200 executed iterations: individual calls may terminate early. The reported comparisons use all 100 group references per setting; the query mask above is not a group-level solver exclusion.

Table 4: Solver settings used to compute the actual PBRF reference points.

Model	Precision / init.	Solver	Budget and stopping
Logistic	FP64; GI initialization	Newton–PCG with strong-Wolfe L-BFGS fallback	PCG up to 100 steps; L-BFGS fallback up to 2,000 steps; gradient tolerance 10^{-10}
MLP	FP64; GI initialization	Strong-Wolfe L-BFGS, learning rate 1, history 50	20 calls, each with up to 10 iterations
ResNet	Original-model initialization	Strong-Wolfe L-BFGS, learning rate 0.5, history 20	100 FP32 warm steps followed by 40 FP64 steps

C.6 E2 IMPLEMENTATION

We use the component directions $\Delta\theta_r$ and coefficients α_r from Equation (16), without changing the partition or recomputing directions during optimization. The following equations hold for each $r = 1, \ldots, k$. For removal ($\epsilon = -1/N$), differentiating the Joint objective in Equation (17) gives the $k \times k$ linear system

$$\sum_{s=1}^{k} \alpha_s^\star \, \Delta\theta_r^\top \left(G - \frac{1}{N} \sum_{i \in \mathcal{S}} J_i^\top H_i J_i \right) \Delta\theta_s = \frac{1}{N} \Delta\theta_r^\top c_{\mathcal{S}}. \tag{27}$$

For Intra, differentiating Equation (19) instead gives

$$\sum_{s=1}^{k} \alpha_s^\star \, \Delta\theta_r^\top G \Delta\theta_s - \frac{\alpha_r^\star}{N} \sum_{i \in \mathcal{C}_r} \Delta\theta_r^\top J_i^\top H_i J_i \Delta\theta_r = \frac{1}{N} \Delta\theta_r^\top c_{\mathcal{S}}. \tag{28}$$

We solve these systems directly after checking positive definiteness and verify the solve residuals, without method-specific damping. Coefficients are unconstrained: no simplex, nonnegativity, or target-dependent fitting is used.

For Joint E2, the parameter-space curvature is the retained-example GGN plus the positive ridge, so the coefficient matrix is positive definite when the component directions are linearly independent; the corresponding property is checked separately for Intra. Intra removes only parts of the edit curvature: the common terms $\Delta\theta_r^\top G \Delta\theta_s$ generally remain nonzero between different components, and the coefficients are still jointly optimized. At $k = 1$, both partition geometries give the same single component; Joint and Intra coincide and only rescale GI. Minimizing either surrogate does not guarantee an improvement against actual PBRF for every group or query.

Full E2. Full E2 minimizes the same Joint quadratic objective over the full parameter space, without restricting the parameter change to the component span. Dogfish uses FP64 PCG with a dense Cholesky fallback. MNIST uses FP64 matrix-free CG with an independently checked relative residual of 10^{-8}. Its linear query readout is evaluated against the same actual PBRF reference. Full E2 is an approximation baseline, not the nonlinear reference or a guaranteed upper bound on prediction accuracy. Table 9 reports the available full-space results.

C.7 COMPONENT-ANALYSIS PROTOCOL

We use the original full-group Bregman $k = 2$ splits of the MNIST 10% groups and order components by size. Contributions are totals over training examples, not per-example averages. For each component, we form a ten-dimensional class profile by averaging its linear influence $\nabla_\theta f_j(\theta_0)^\top \Delta\theta_r$ within each test class, using the full 10,000-example test set. Cosine similarity compares the two ten-dimensional profiles within each group; the illustrated profiles are unit-normalized. These class averages are distinct from the 100 individual-query targets used to score prediction accuracy.

Omission retains the full-group PBRF reference and sets one component contribution to zero. GI assigns unit weight to both components; single-component E2 ($k = 1$) applies the same optimized weight to both; $k = 2$ Joint E2 uses the two optimized component weights. Large-only and small-only predictions retain the corresponding contribution at its full-fit weight and set the other to zero. We neither refit the remaining coefficient nor change the removed set to the retained component. Appendix D.4 reports the class-pattern case and the full/large-only/small-only prediction results.

C.8 Computational complexity and timing protocol

Let m be group size, d parameter dimension, k component count, t a bound on the total Lloyd iterations across restarts, $\mathcal{C}_{\text{inv}}$ the cost of an inverse-G solve, and γ the per-example cost of a directional GGN product. We assume additive work over examples when counting GGN products. We separate partition construction, component-response solves, edit curvature, and the coefficient solve.

Partition construction. Given explicit gradients, a center-contrast implementation has costs

$$T_{\text{Bregman,partition}} = O\big(t\big[(k-1)\mathcal{C}_{\text{inv}} + mkd\big]\big)\,,$$
$$T_{\text{Euclidean,partition}} = O(tmkd). \tag{29}$$

These expressions exclude gradient construction and shared model/data preparation. Euclidean clustering removes the $k-1$ center-contrast metric solves per Lloyd iteration, not the inverse solves needed to form the final component responses. The matrix-free implementation evaluates the same geometry through products rather than forming a $d \times d$ inverse. For dense Dogfish, a Cholesky factorization of G costs $O(d^3)$ once per checkpoint, whitening m gradients costs $O(md^2)$, and Lloyd work costs $O(tmkd)$. Euclidean clustering omits the metric preparation/solves, but the original G is still used after partitioning.

Correction at fixed directions. Joint evaluates each of the k directions over all m removed examples, whereas Intra evaluates an example only along its own component's direction. With vector-valued products and coefficient-space projection, the edit costs are

$$T_{\text{Joint,edit}} = O(km\gamma + dk^2), \qquad T_{\text{Intra,edit}} = O(m\gamma + dk). \tag{30}$$

The example–direction count drops from km to $\sum_r |\mathcal{C}_r| = m$, regardless of component balance. Both methods retain the global component-response solves, with cost $O(k\mathcal{C}_{\text{inv}})$, the common matrix with entries $\Delta\theta_r^\top G \Delta\theta_s$, and a dense $k \times k$ coefficient solve costing $O(k^3)$. Forming the common matrix from scratch requires k applications of G and $O(dk^2)$ inner-product work; cached identities may reduce this work. Intra does not turn G into smaller parameter-space blocks. Its edit cost is of the same order as single-component E2 ($k = 1$), so these expressions alone do not imply an end-to-end speedup over $k = 1$.

Partition timing. We use 100 MNIST 1% groups at $k = 2$, FP64, with each method pair on the same GPU: 82 A100-SXM4-80GB, five A100-SXM4-40GB, and 13 H200 NVL pairs. Each method is timed once per group; Bregman precedes Euclidean, without dedicated timing warm-up. Both use two restarts and unchanged-assignment stopping, with a 100-iteration cap per restart. Both restarts converge for every group. The total iterations across restarts have a median of nine for each method, with ranges of 6–12 for Bregman and 6–16 for Euclidean. The wall-clock interval includes group-gradient preparation, metric solves within clustering, Lloyd assignments, restart selection, and progress serialization. Shared model/data/operator construction, baseline validation, final component-response solves, E2, and query readout are excluded. Shared preparation is neither timed nor amortized into these values. Scalar and CPU operations synchronize within the computation; no extra CUDA synchronization was placed at the timer boundaries. These are paired stage measurements over different groups, not repeated cold-start benchmarks.

Edit-curvature timing. Table 14 uses the 49 matched MNIST 10% groups executed on an RTX3090 in FP64, with fixed Euclidean partitions and cached directions. Each method is timed once per group in the existing worker order, rather than in independently warmed-up repeated trials. The Joint interval excludes coefficient-space projection; the Intra interval includes scalar direction products and auxiliary vector accumulation. Both exclude partition construction, response inverse solves, the common full-source Gram computation, the coefficient solve, and final query readout.

The intervals therefore document the instrumented edit stages, not identical low-level instruction sets or end-to-end runtimes. Appendix D.5 reports the measured partition and edit-curvature costs under these protocols.

D SUPPLEMENTARY EXPERIMENTAL RESULTS

We distinguish the error diagnostic from prediction accuracy, and partition geometry from the curvature retained in the response and correction. All measured results use the protocol in Appendix C; table entries are query-wise medians unless stated otherwise.

D.1 GROUP-SIZE ABLATION

BI and squared GI error. Table 5 varies the removal ratio over $1, 2, 5, 10, 20, 35, 50\%$ for all three models. Each ratio uses 100 random groups and the same 100 queries within a model's sweep. For every query, we correlate projected BI with squared GI error across groups, then summarize the 100 correlations by their median. The association weakens with removal size: between 1% and 50%, the median decreases from 0.865 to 0.140 on Dogfish, from 0.701 to 0.121 on MNIST, and from 0.525 to 0.145 on ResNet9. These diagnostic correlations are not the prediction–reference correlations used to evaluate correction methods.

Table 5: Removal-ratio ablation for projected BI versus squared GI error. Each entry is the median of 100 query-wise Spearman correlations, each computed across 100 random groups. The original model and query bank are fixed within each model's size sweep.

Model	1%	2%	5%	10%	20%	35%	50%
Dogfish logistic	0.865	0.815	0.676	0.549	0.475	0.327	0.140
MNIST-10 MLP	0.701	0.656	0.580	0.453	0.340	0.169	0.121
CIFAR-10 ResNet9	0.525	0.413	0.290	0.218	0.190	0.146	0.145

Correction accuracy and paired comparisons. Table 6 compares single-component E2 ($k = 1$) with Euclidean Joint and Intra E2 ($k = 2$) for the available removal ratios. Dogfish and MNIST cover $1, 10, 30\%$; ResNet9's correction comparison is at 1%. The Dogfish 1% Intra comparison is unavailable and is not filled by a result from another setting. The group and query banks are matched across methods within each setting; the diagnostic and correction sweeps are separate experiments. The paired tests follow Appendix C.3 and use all common queries, not a selected representative query.

On MNIST at 10%, both Joint and Intra improve raw R^2 significantly in this query-paired analysis, although the Intra median gain is only 0.000635. Statistical significance therefore should not be confused with the magnitude of the improvement. At MNIST 30%, Intra instead decreases raw R^2 significantly while improving Spearman, separating value accuracy from ranking accuracy.

Table 7 includes GI and additional $k = 10$ comparisons at these supplementary sizes. For MNIST at 30%, Euclidean Intra's median R^2 is below single-component E2 ($k = 1$) at both $k = 2$ and $k = 10$, while Joint improves as more directions are available. This is an observed accuracy–cost trade-off, not a guarantee that either approximation is uniformly preferable.

D.2 GEOMETRY ABLATION AND THE EUCLIDEAN APPROXIMATION

Three considerations motivate the Euclidean approximation: similar associations between dispersion and GI error, similar correction accuracy across partition geometries, and precedents in gradient-based attribution.

Similar dispersion–error associations. Table 8 compares projected BI, full BI, and raw-gradient Euclidean dispersion using the same Dogfish and MNIST 1% groups and query-specific squared GI errors. Full BI and Euclidean dispersion yield similar median correlations: 0.759 versus 0.760 on Dogfish and 0.556 versus 0.558 on MNIST. Projected BI instead reaches 0.865 and 0.701, respectively. Thus, replacing the curvature-weighted dispersion with its Euclidean counterpart has little effect on the observed error association, whereas target projection provides additional query-specific information. This comparison concerns dispersion measures; the effect of partition geometry on correction accuracy is examined next.

Table 6: Median query-wise accuracy and two-sided query-paired p-values for Euclidean E2 ($k = 2$) versus single-component E2 ($k = 1$). Holm correction covers all 26 comparisons jointly across settings, variants, and both metrics (Appendix C.3). The Spearman tests compare methods and are not tests of zero correlation. MNIST 30% Intra shows a significant decrease in raw R^2, not an improvement.

Removal	Metric	$k = 1$	Joint $k = 2$	p_{adj}	Intra $k = 2$	p_{adj}
Dogfish						
1%	R^2	0.995	0.999	1.01×10^{-16}	—	
	ρ	0.998	1.00	1.35×10^{-16}	—	
10%	R^2	0.946	0.985	1.01×10^{-16}	0.982	1.01×10^{-16}
	ρ	0.985	0.994	1.01×10^{-16}	0.993	1.15×10^{-16}
30%	R^2	0.891	0.945	1.40×10^{-16}	0.945	1.65×10^{-15}
	ρ	0.981	0.989	1.36×10^{-15}	0.984	2.87×10^{-5}
MNIST						
1%	R^2	0.998	1.00	2.83×10^{-16}	1.00	3.38×10^{-16}
	ρ	0.999	1.00	1.02×10^{-16}	1.00	1.02×10^{-16}
10%	R^2	0.993	0.995	5.64×10^{-15}	0.994	2.22×10^{-6}
	ρ	0.997	0.998	3.16×10^{-15}	0.998	1.24×10^{-14}
30%	R^2	0.964	0.968	4.23×10^{-9}	0.962	2.34×10^{-4}
	ρ	0.992	0.993	3.46×10^{-7}	0.992	1.96×10^{-4}
ResNet9						
1%	R^2	0.965	0.975	2.83×10^{-16}	0.974	2.83×10^{-16}
	ρ	0.986	0.989	7.27×10^{-16}	0.989	5.37×10^{-16}

Table 7: Additional removal ratios, with 100 random groups and 100 retained fixed queries per setting. All methods within a setting share the checkpoint, groups, queries, and actual PBRF references. The selected $k = 2$ and $k = 10$ comparisons expose removal-size and component-count sensitivity.

Method	Spearman	R^2
Dogfish (10%)		
GI	0.988	0.901
Single-component E2 ($k = 1$)	0.985	0.946
Euclidean Joint E2 ($k = 2$)	0.994	0.985
Euclidean Intra E2 ($k = 2$)	0.993	0.982
Euclidean Joint E2 ($k = 10$)	1.00	0.997
Euclidean Intra E2 ($k = 10$)	0.990	0.981
MNIST (1%)		
GI	0.999	0.990
Single-component E2 ($k = 1$)	0.999	0.998
Euclidean Joint E2 ($k = 2$)	1.00	1.00
Euclidean Intra E2 ($k = 2$)	1.00	1.00
Euclidean Joint E2 ($k = 10$)	1.00	1.00
Euclidean Intra E2 ($k = 10$)	0.999	0.999
MNIST (30%)		
GI	0.992	0.921
Single-component E2 ($k = 1$)	0.992	0.964
Euclidean Joint E2 ($k = 2$)	0.993	0.968
Euclidean Intra E2 ($k = 2$)	0.992	0.962
Euclidean Joint E2 ($k = 10$)	0.995	0.977
Euclidean Intra E2 ($k = 10$)	0.990	0.953

Table 8: Geometry ablation on the same random 1% groups and fixed queries as the BI experiment. Entries are medians over 100 query-wise Spearman correlations with squared GI error, each computed across 100 groups. Euclidean dispersion uses raw gradients, not parameter responses.

Model	Projected BI	Full BI	Euclidean
Dogfish logistic	0.865	0.759	0.760
MNIST-10 MLP	0.701	0.556	0.558

Similar correction accuracy. Table 1 compares Bregman and Euclidean partitions under both Joint and Intra E2 at $k = 2$ on Dogfish 30% and MNIST 10% groups. Only the partition metric changes; component responses and coefficient fitting retain the original G. The two geometries yield similar prediction accuracy under either curvature scope. For Joint E2, median raw R^2 rounds to 0.945 on Dogfish and 0.995 on MNIST for both geometries. Tables 10 to 12 extend this comparison across component counts. These results support the approximation empirically rather than establishing equivalence between the partitions.

Precedents in gradient-based attribution. Gradient-inner-product attribution provides a related precedent for omitting curvature weighting. TracIn (Pruthi et al., 2020) approximates training-induced loss changes using gradient inner products accumulated along training checkpoints. At a single checkpoint, this inner-product form is related algebraically to replacing inverse curvature by the identity. The full trajectory-based method is not identical to that substitution in classical influence functions. In our method, the identity metric is used only to construct the partition; component responses and coefficient fitting still use the original G.

Together, these considerations motivate Euclidean clustering as a practical partition approximation without requiring G to be close to the identity or the two geometries to produce identical assignments. The resulting partition-construction savings are reported in Appendix D.5, with complexity and measurement protocols in Appendix C.8.

D.3 Component count and Joint–Intra correction

Rescaling, reweighting, and the full-space comparison. Table 9 collects the main correction settings: Dogfish at 30% removal, MNIST at 10%, and ResNet9 at 1%. The $k = 1$ comparison isolates rescaling along the original GI direction; $k > 1$ also permits changes in relative component contributions. At $k = 1$, both geometries give the same single component, and Joint and Intra E2 coincide. Full E2 removes the component-span restriction but retains the same quadratic edit approximation. Its stronger results on Dogfish and MNIST indicate room for improvement beyond the restricted response space; they do not make Full E2 equivalent to actual PBRF.

Complete component-count sweeps. Tables 10 and 11 report $k \in \{1, 2, 4, 6, 8, 10\}$ on the two main correction cohorts. Table 12 additionally reports Dogfish at 10% removal, allowing comparison with MNIST at the same removal ratio. For Dogfish 10%, median raw R^2 increases from 0.901 for GI to 0.946 for $k = 1$ E2 and 0.985 for Bregman Joint E2 at $k = 2$; Euclidean Intra E2 at $k = 2$ reaches 0.982. For MNIST 10%, the corresponding Bregman Joint improvement is smaller, from 0.993 at $k = 1$ to 0.995 at $k = 2$. Bregman and Euclidean denote only the partition metric; both retain the full G in component responses and coefficient fitting. Joint and Intra differ only in the additional edit-curvature term, as specified in Equations (27) and (28).

Euclidean and Bregman Joint E2 have similar accuracy in these settings. Intra saves edit-curvature work but need not preserve Joint's accuracy: its raw R^2 decreases at larger k in the reported cohorts. The results therefore do not establish that inter-component edit curvature is universally negligible or that increasing k always improves the approximation. Partitions obtained at different k need not define nested response spaces.

D.4 Why retain and reweight both components?

Distinct class-wise patterns. Using the profiles defined in Appendix C.7, the displayed MNIST 10% group has component sizes 5,997 and 3 and cosine similarity -0.457. Across all 100 groups,

Table 9: Correction results against actual PBRF reference points. All settings use 100 random groups; metrics are medians over the common retained queries (100 for Dogfish and MNIST, and 97 for ResNet9). All methods within a setting share the checkpoint, groups, queries, and actual PBRF references. Full E2 is an additional approximation, not the reference itself.

Method	Spearman	R^2
Dogfish (30%)		
GI	0.984	0.862
Single-component E2 ($k = 1$)	0.981	0.891
Bregman Joint E2 ($k = 2$)	0.989	0.945
Bregman Intra E2 ($k = 2$)	0.984	0.945
Euclidean Joint E2 ($k = 2$)	0.989	0.945
Euclidean Intra E2 ($k = 2$)	0.984	0.945
Full E2	1.00	0.990
MNIST (10%)		
GI	0.997	0.978
Single-component E2 ($k = 1$)	0.997	0.993
Bregman Joint E2 ($k = 2$)	0.998	0.995
Bregman Intra E2 ($k = 2$)	0.998	0.993
Euclidean Joint E2 ($k = 2$)	0.998	0.995
Euclidean Intra E2 ($k = 2$)	0.998	0.994
Full E2	0.999	0.999
ResNet9 (1%)		
GI	0.986	0.891
Single-component E2 ($k = 1$)	0.986	0.965
Euclidean Joint E2 ($k = 2$)	0.989	0.975
Euclidean Intra E2 ($k = 2$)	0.989	0.974

Table 10: Dogfish, 30%: component-count sweep on the same 100 groups and 100 queries. Each entry reports the median over fixed test queries. The four variants coincide at $k = 1$.

	Spearman ρ				R^2			
	Joint E2		Intra E2		Joint E2		Intra E2	
k	Bregman	Euclidean	Bregman	Euclidean	Bregman	Euclidean	Bregman	Euclidean
1	0.981	0.981	0.981	0.981	0.891	0.891	0.891	0.891
2	0.989	0.989	0.984	0.984	0.945	0.945	0.945	0.945
4	0.993	0.993	0.984	0.984	0.968	0.970	0.951	0.950
6	0.995	0.996	0.982	0.982	0.978	0.978	0.946	0.947
8	0.996	0.996	0.980	0.979	0.981	0.981	0.941	0.941
10	0.997	0.997	0.977	0.978	0.983	0.983	0.939	0.940

Table 11: MNIST, 10%: component-count sweep on the same 100 groups and 100 queries. Each entry reports the median over fixed test queries. The four variants coincide at $k = 1$.

	Spearman ρ				R^2			
	Joint E2		Intra E2		Joint E2		Intra E2	
k	Bregman	Euclidean	Bregman	Euclidean	Bregman	Euclidean	Bregman	Euclidean
1	0.997	0.997	0.997	0.997	0.993	0.993	0.993	0.993
2	0.998	0.998	0.998	0.998	0.995	0.995	0.993	0.994
4	0.998	0.998	0.998	0.998	0.996	0.996	0.994	0.994
6	0.999	0.999	0.998	0.998	0.996	0.997	0.994	0.994
8	0.999	0.999	0.998	0.998	0.997	0.997	0.993	0.994
10	0.999	0.999	0.998	0.998	0.997	0.997	0.993	0.993

Table 12: Dogfish, 10%: component-count sweep on the same 100 groups and 100 queries. Each entry reports the median over fixed test queries. The four variants coincide at $k = 1$.

	Spearman ρ				R^2			
	Joint E2		Intra E2		Joint E2		Intra E2	
k	Bregman	Euclidean	Bregman	Euclidean	Bregman	Euclidean	Bregman	Euclidean
1	0.985	0.985	0.985	0.985	0.946	0.946	0.946	0.946
2	0.994	0.994	0.993	0.993	0.985	0.985	0.982	0.982
4	0.998	0.998	0.994	0.994	0.995	0.996	0.986	0.987
6	0.999	0.999	0.993	0.994	0.997	0.997	0.984	0.985
8	0.999	0.999	0.992	0.992	0.997	0.997	0.983	0.983
10	1.00	1.00	0.991	0.990	0.997	0.997	0.981	0.981

the median profile cosine is -0.027, the interquartile range is $[-0.250, 0.165]$, and the 90th percentile is 0.359. Component size alone therefore does not describe its class-wise influence pattern.

Omission against the full-group reference. Table 13 retains both components, only the larger component, or only the smaller component, using the same full-group actual PBRF reference and fixed full-fit coefficients. For Bregman Joint E2 ($k = 2$), median raw R^2 falls from 0.995 with both components to 0.636 with large-only and -0.009 with small-only. The full-versus-large-only comparison demonstrates the predictive contribution of the small component; full versus small-only demonstrates that of the large component. The poor small-only prediction is not evidence that the small component can be discarded. The same qualitative degradation occurs for GI and single-component E2 ($k = 1$). These results motivate retaining and reweighting both components rather than dropping one based on size. They do not test a refitted one-component model or establish independence between components.

Table 13: MNIST 10%, Bregman Joint E2 ($k = 2$): component omission against the same full-group PBRF reference using 100 groups and 100 queries. E2 coefficients are held at the full-fit values; they are not refitted after omission.

Coefficient model	Retained contributions	Spearman	R^2
GI	Full	0.997	0.978
	Large only	0.794	0.630
	Small only	0.492	-0.008
Single-component E2 ($k = 1$)	Full	0.997	0.993
	Large only	0.797	0.638
	Small only	0.492	-0.002
Bregman Joint E2 ($k = 2$)	Full	0.998	0.995
	Large only	0.797	0.636
	Small only	0.493	-0.009

D.5 Stage-wise computational savings

The costs in Equations (29) and (30) concern different stages; the measurement boundaries and hardware are specified in Appendix C.8. For MNIST 1% at $k = 2$, median partition-construction time decreases from 59.4 seconds for Bregman to 0.277 seconds for Euclidean clustering, a 99.5% reduction from the two medians over 100 matched groups. This comparison excludes shared preparation and final component-response solves.

Table 14 reports the instrumented edit-curvature intervals for the 49 matched MNIST 10% groups with fixed Euclidean directions on an RTX3090. At $k = 2$, Joint and Intra have medians of 0.193 and 0.103 seconds, respectively, a 46.6% reduction. The common inverse solves, full-source Gram computation, and coefficient solve are excluded from these intervals. The measurements support lower cost for the replaced stages at fixed k, not an end-to-end speedup over single-component E2 ($k = 1$).

Table 14: Median edit-curvature stage time (seconds), MNIST 10%, fixed Euclidean components, 49 matched groups on an RTX3090. Reductions use the two medians. These are instrumented stage intervals, not end-to-end timings.

k	Joint	Intra	Reduction (%)
2	0.193	0.103	46.6
4	0.384	0.111	71.2
10	0.973	0.136	86.0