Distributionally Robust Data Valuation

Xiaoqiang Lin [1] **Xinyi Xu** [1] **Zhaoxuan Wu** [2] **See-Kiong Ng** [2] **Bryan Kian Hsiang Low** [1]

Abstract

Data valuation quantifies the contribution of each data point to the performance of a machine learning model. Existing works typically define the value of data by its improvement of the validation performance of the trained model. However, this approach can be impractical to apply in collaborative machine learning and data marketplace since it is difficult for the parties/buyers to agree on a common validation dataset or determine the exact validation distribution *a priori*. To address this, we propose a *distributionally robust data valuation* approach to perform data valuation without known/fixed validation distributions. Our approach defines the value of data by its improvement of the distributionally robust generalization error (DRGE), thus providing a worst-case performance guarantee *without* a known/fixed validation distribution. However, since computing DRGE directly is infeasible, we propose using *model deviation* as a proxy for the marginal improvement of DRGE (for kernel regression and neural networks) to compute data values. Furthermore, we identify a notion of uniqueness where low uniqueness characterizes low-value data. We empirically demonstrate that our approach outperforms existing data valuation approaches in data selection and data removal tasks on real-world datasets (e.g., housing price prediction, diabetes hospitalization prediction).

1. Introduction

In machine learning (ML), data is essential to obtaining good learning performance. Data valuation (Jia et al., 2019; Ghorbani & Zou, 2019; Ghorbani et al., 2020; Sim et al., 2020) is introduced to quantify the contribution (i.e., value) of each data point to the model performance. The resultant data values are useful in many ways. For example, in data marketplaces (Yu & Zhang, 2017; Agarwal et al., 2019), data values can be used to price data. In collaborative machine learning (CML), data values can be used to determine the participating parties' contributions to the collaboration/training and fairly reward them (e.g., monetary rewards) (Sim et al., 2020; Wang et al., 2020; Lin et al., 2024). Data valuation also finds applications in explainable machine learning (e.g., dataset debugging) (Koh & Liang, 2017) and active learning (Ghorbani et al., 2021). Of note, existing works (Ghorbani & Zou, 2019; Jia et al., 2019; Kwon & Zou, 2021; Wang & Jia, 2023) typically define the value of data based on its improvement of validation performance.

However, using the performance on a specific validation dataset for data valuation can be impractical in some cases. Specifically, in CML and data marketplace, ensuring all parties/buyers agree on the same validation dataset is difficult since they usually have heterogeneous local data and validation distributions (Li et al., 2021; Xu et al., 2021). For instance, hospitals collaboratively collect data to train a model for hospitalization prediction. The value of each dataset (from different hospitals) depends on the unknown distribution of future model users (Walker, 2015; Tu et al., 2022), as new hospitals with varying local validation distributions may join later. Therefore, valuing data based on a specific validation dataset can misrepresent the data's value on the validation distribution of such a hospital that joins later. Separately, in data marketplaces (e.g., AWS Data Exchange (Amazon, 2023)), sellers need to price the data before knowing exactly who the buyers are (Just et al., 2023), meaning the sellers do not know the local validation distributions of these buyers *a priori* (i.e., no known validation distribution). In this scenario, existing approaches that require a known validation distribution (Jia et al., 2019; Wang et al., 2021; Wu et al., 2022) cannot be applied. The core question is: *What perspective should we take to value data without a fixed and known validation dataset/distribution?*

Although without a known/fixed validation distribution, it is still desirable if the data values can indicate some performance guarantee. Specifically, the buyers in the data marketplace (or parties in CML) are still interested in how the data can benefit them in improving the model performance on

[1]Department of Computer Science, National University of Singapore, Singapore [2]Institute of Data Science, National University of Singapore, Singapore. Correspondence to: Zhaoxuan Wu <wu.zhaoxuan@u.nus.edu>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

their respective local data (i.e., higher data value indicates the data point can lead to higher performance improvement). Fortunately, we can draw a parallel to distributionally robust optimization (Hu et al., 2018; Staib & Jegelka, 2019; Rahimian & Mehrotra, 2019) where an ML model is trained to minimize the worst-case performance over an uncertainty set of validation distributions defined as an ε-ball around a reference distribution. In our scenario where sellers must price data without knowing the validation distribution, we define a data value using the *distributionally robust generalization error* (DRGE), namely the worst-case performance under a set of distributions. Consequently, if the validation distribution of one buyer is within the uncertainty set, the validation performance is guaranteed to be better than DRGE. This definition means that we evaluate the value of data points based on how much they improve over DRGE, which can thus indicate the (lower bound of) performance improvement for a potential buyer if the buyer's validation distribution is within the uncertainty set.

Despite the appeal, unfortunately, it is difficult to use directly DRGE as the so-called utility function for data valuation due to the lack of an analytical form for DRGE (Sec. 2). Fortunately, in data valuation, the analytic form of the utility function (i.e., DRGE) is often *not* needed; instead the marginal improvement of the utility function (i.e., DRGE) is sufficient. Specifically, existing data valuation approaches (Jia et al., 2019; Ghorbani & Zou, 2019; Kwon & Zou, 2021) using leave-one-out (LOO), Shapley value, or semivalue, all compute data values via a weighted average of marginal improvements of the utility function. For example, LOO uses the marginal improvement of utility gained by adding a data point to the rest of the dataset. Therefore, knowing the marginal improvement of a data point w.r.t. the specified utility function is sufficient to compute the data values. Based on this observation, we adopt the *model deviation* as the proxy for the marginal improvement of a data point on DRGE. Model deviation quantifies the discrepancy between a model trained on a dataset without a specific data point and one trained on the complete dataset, including that point. We leverage the reproducing kernel Hilbert space (RKHS) to formalize model deviation for kernel-based algorithms.

As a result, our approach's definition of model deviation requires RKHS models, making it difficult to apply to non-RKHS ones, in particular neural networks (NNs), which are widely used in CML and data marketplaces (McMahan et al., 2017; Wu et al., 2022). To resolve this difficulty, we utilize the neural tangent kernel (NTK) theory to define and efficiently compute model deviation for NN. Specifically, we leverage a theoretical connection between the predictions of a wide NN and a kernel regression with NTK (Arora et al., 2019) to show that the model deviation computed with NTK is a suitable proxy for the marginal improvement of DRGE of NN. By incorporating DRGE and model deviation, we introduce the distributionally robust data valuation framework to answer the core question earlier. That is, when a fixed and known validation dataset is not available, we adopt the DRGE to define the value of data. Due to the computational intractability of DRGE, we use model deviation to compute data values. We demonstrate that model deviation is an appropriate proxy for the marginal improvement of DRGE in the context of a kernel-based algorithm, which is sufficient for data valuation. Moreover, we leverage the NTK theory to extend the result to NN and propose a computationally efficient approach to compute model deviation, thus making our approach more widely applicable (than only RKHS models). Interestingly, a notion of uniqueness arises from our framework that characterizes what data points lead to low model deviation and thus low data value.

To summarize, we have the following contributions:

- Defining a *distributionally robust generalization error* to provide a novel perspective of data valuation *without* known or fixed validation distributions.
- Proposing *model deviation* as a proxy for the marginal improvement of DRGE to resolve the intractability of DRGE in computing data value for kernel-based algorithms.
- Leveraging the NTK theory to show that model deviation with NTK can be a proxy for marginal improvement of DRGE of NN and proposing an approach to compute model deviation efficiently.
- Identifying a notion of uniqueness that consists of *scarcity* and *dissimilarity*. Lower scarcity or dissimilarity leads to lower uniqueness, model deviation, and data value.
- Empirically demonstrating that our approach outperforms other data valuation approaches in applications of data values, specifically in data removal and subset selection.

2. Setting and Preliminaries

Data valuation. We consider the supervised learning setting and denote data by $x \in \mathcal{X} \subseteq \mathbb{R}^d$. Each data has a label $y \in \mathcal{Y}$, obtained from a labeling function $f^* : \mathcal{X} \to \mathcal{Y}$ (i.e., the true label for x is $f^*(x)$). Denote $N \coloneqq \{1, \ldots, n\}$ and a training dataset $D_N \coloneqq \{z_j = (x_j, y_j)\}_{j=1}^n$ where $\{x_j\}_{j=1}^n$ are i.i.d. sampled from the sampling distribution P defined on $\mathcal{X}$. Denote $S \subseteq N$ and its corresponding dataset $D_S \coloneqq \{(x_j, y_j); j \in S\}$.

Existing works (Ghorbani & Zou, 2019; Jia et al., 2019; Kwon & Zou, 2021; Wang & Jia, 2023) define the data value for data point z_i as the weighted average of the marginal improvements of the data point when added to different data subsets, such as LOO, Shapley value, and Banzhaf value. Formally, the marginal improvement of the data point z_i to the data subset $D_S \subseteq D_N \setminus \{z_i\}$ is defined as:

$$\Delta_{i,S} \coloneqq U(S \cup \{i\}) - U(S) \tag{1}$$

where $U : 2^N \to \mathbb{R}$ is the utility function that quantify the utility of D_S. Most data valuation works define $U(S)$ as the validation performance (Jia et al., 2019; Kwon & Zou, 2021; Wu et al., 2022) (w.r.t. a validation distribution) of the model trained on D_S. The data value of z_i is defined as follows:

$$\varphi_i \coloneqq \textstyle\sum_{S\subseteq N\setminus\{i\}} w(S)\Delta_{i,S} \tag{2}$$

where $w : 2^N \to \mathbb{R}$ is the function that gives a weight to each set S. LOO specifies the $w(S) = \mathbb{1}(S = N \setminus \{i\})$, Shapley value specifies $w(S) = |S|!(n-|S|-1)!/n!$, and Banzhaf value specifies $w(S) = 1/2^{n-1}$.

Kernel-based algorithms. Denote $\mathcal{H}$ as an RKHS with inner product defined as $\langle\cdot,\cdot\rangle_{\mathcal{H}}$ and norm as $\|\cdot\|_{\mathcal{H}} = \sqrt{\langle\cdot,\cdot\rangle_{\mathcal{H}}}$. Denote a mapping function $\Phi : \mathcal{X} \mapsto \mathcal{H}$ and a kernel function $K : \mathcal{X}\times\mathcal{X} \to \mathbb{R}$ such that $K(x_1, x_2) = \langle\Phi(x_1), \Phi(x_2)\rangle_{\mathcal{H}}$. We use $\mathcal{H}_K$ to denote the RKHS associated with kernel K when a kernel is specified. A kernel-based algorithm can be cast as the minimization problem $\operatorname{argmin}_{f\in\mathcal{H}_K} \sum_{(x_i,y_i)\in D_S} \ell\big(f(x_i), y_i\big)+G(\|f\|_{\mathcal{H}_K})$ where ℓ is the loss function and G is the regularization function. As an example, for kernel ridge regression: $\ell\big(f(x_i), y_i\big)= (1/2)\big(f(x)_i - y_i\big)^2$ and $G(\|f\|_{\mathcal{H}_K}) = (1/2)\lambda\|f\|^2_{\mathcal{H}_K}$ where λ is the regularization parameter. The minimization problem for kernel ridge regression has a closed-form solution $f_S = \big((K_S + \lambda I)^{-1}\mathcal{Y}_S\big)^\top K(\mathcal{X}_S, \cdot)$, where K_S is the kernel matrix with $[K_S]_{j,k} = K(x_j, x_k), \forall x_j, x_k \in D_S$ and $\mathcal{Y}_D = [y_1, \ldots, y_{|D_S|}]^\top$ and $K(\mathcal{X}_S, \cdot) = [K(x_1, \cdot)^\top, \ldots, K(x_{|D_S|}, \cdot)^\top]^\top$. We further denote $K_{S,S\cup\{i\}}$ as the kernel matrix with $[K_{S,S\cup\{i\}}]_{j,k} = K(x_j, x_k), \forall x_j \in D_S, x_k \in D_{S\cup\{i\}}$.

Neural tangent kernel (NTK). Define a fully-connected NN f parameterized by θ as follows,

$$f^{(m)}(x) \coloneqq W^{(m)} g^{(m-1)}(x) \in \mathbb{R}^{d_m},$$
$$g^{(m)}(x) \coloneqq \sqrt{c_\sigma/d_m}\,\sigma\big(f^{(m)}(x)\big) \in \mathbb{R}^{d_m}, m = 1, \ldots, L_{\text{NN}}$$

where $g^0(x) = x$ and d_0 is the input dimension. The $W^m \in \mathbb{R}^{d_m\times d_{m-1}}$ is the weight matrix in the m-th layer and $\sigma(\cdot)$ is the activation function, $c_\sigma = (\mathbb{E}_{z\sim\mathcal{N}(0,1)}[\sigma(z)^2])^{-1}$. The output of the NN is $f(x) \coloneqq W^{(L_{\text{NN}}+1)} g^{(L_{\text{NN}})}(x)$. Each NN parameter is initialized with an i.i.d. sample from $\mathcal{N}(0, 1)$. Denote the initialized θ as θ_0. The NTK at initialization is defined as $\Theta_0(x_i, x_j) \coloneqq \nabla_{\theta=\theta_0} f(x_i)^\top \nabla_{\theta=\theta_0} f(x_j)$ for $x_i, x_j \in \mathcal{X}$ (Arora et al., 2019). As each of $d_1, \ldots, d_{L_{\text{NN}}} \to \infty$, Θ_0 will converge to a deterministic form Θ (namely the theoretical NTK) (Lee et al., 2019). A fully trained infinite-width NN with gradient descent is equivalent to kernel regression predictor with Θ (Arora et al., 2019). Moreover, the difference between the outputs of a sufficiently wide finite-width converged NN and kernel regression with Θ is small (Arora et al., 2019). The NTK is also widely used in other settings, e.g., active learning for NNs (Hemachandra et al., 2023; Lau et al., 2024). Define NTK matrix Θ_S similarly as kernel matrix K_S and denote $\mathcal{H}_\Theta$ as the RKHS of Θ.

3. Distributionally Robust Data Valuation

We present our data valuation approach for the setting of no known/fixed validation distribution. We propose using DRGE as the utility function and show that access to its marginal improvement is sufficient for data valuation in Sec. 3.1. In Sec. 3.2, we introduce model deviation as a proxy for DRGE's marginal improvement. In Sec. 3.3, we extend the model deviation to NN using NTK theory and provide an efficient closed-form computation of model deviation.

3.1. Distributionally Robust Generalization Error

The empirical risk of a model f under one specific distribution P is defined as $\mathbb{E}_{x\sim P}\big(\ell(f(x), f^*(x))\big)$ where ℓ is the loss function. When the validation distribution is unknown, the empirical risk of a specific distribution P is less insightful since it does not reflect model performance when validation distributions differ from P (Zhang et al., 2022), motivating the following definition:

Definition 3.1. For a sampling distribution P, the distributionally robust generalization error (DRGE) is

$$R(f, \mathcal{Q}) \coloneqq \sup_{Q\in\mathcal{Q}} \mathbb{E}_{x\sim Q}\Big(\ell\big(f(x), f^*(x)\big)\Big) \tag{3}$$

where $\mathcal{Q} \coloneqq \{Q : \chi^2(Q, P) \le \varepsilon\}$ is a set of distributions defined on $\mathcal{X}$ in the ε-ball around P w.r.t. the χ^2-divergence. Specifically, $\chi^2(Q, P) \coloneqq \int_{x\in\mathcal{X}} \big(\frac{q(x)}{p(x)} - 1\big)^2 p(x)dx$ where $p(x)$ and $q(x)$ are probability density functions of distributions P and Q respectively and we assume that Q is absolutely continuous w.r.t. P.

Existing works (Wu et al., 2022; Just et al., 2023) assume different buyers/parties have a common and fixed validation distribution. We relax this assumption to that they have possibly *different* validation distributions, but close to P. Specifically, each of their validation distributions is within ε χ^2-divergence to P (i.e., in $\mathcal{Q}$). Consequently, DRGE is a worst-case performance of a model f w.r.t. all (validation) distributions in $\mathcal{Q}$. We choose the χ^2-divergence to define $\mathcal{Q}$ here because it is applicable to real-world applications. Specifically, it measures the discrepancy between the densities of different subregions in the two compared distributions which is the discrepancy commonly seen in data marketplace applications where buyers' local data distribution is confined to some subregions due to the limited

observations (e.g., having data only for a certain group of users due to limited number of users in a company). Additionally, the simple form of χ^2-divergence is amenable to our theoretical results later. More discussion on our choice of χ^2-divergence and definition of $\mathcal{Q}$ can be found in Appendix A where we compare with other existing definitions. Note that we can relax the Definition 3.1 to consider non-deterministic labels (instead of the deterministic one we have one) by replacing $f^*(x)$ to $\mathbb{E}(y \mid x)$ and all our theoretical results in the following sections will still hold.

Unfortunately, DRGE cannot be directly computed due to the lack of access to distribution P and ε in Definition 3.1. Fortunately, one key observation is that the exact DRGE is *not* needed for data valuation. Specifically, by replacing the utility function in Equ. (1) with $U^{\text{DR}}(D_S) \coloneqq -R(f_S, \mathcal{Q})$, we define the following:

Definition 3.2. The distributionally robust data value is:

$$\varphi_i^{\text{DR}} \coloneqq w(S)\Delta_{i,S}^{\text{DR}}, \quad \Delta_{i,S}^{\text{DR}} \coloneqq R(f_S, \mathcal{Q}) - R(f_{S\cup\{i\}}, \mathcal{Q}) \, .$$

From Definition 3.2, the data value φ_i^{DR} is only dependent on the marginal improvement of DRGE $\Delta_{i,S}^{\text{DR}}$. Hence, being able to compute the marginal improvement of DRGE is sufficient for data valuation. We mainly consider the LOO valuation scheme and hence $\varphi_i^{\text{DR}} = \sum_{S\subseteq N\setminus\{i\}} \mathbb{1}(S = N \setminus \{i\})\Delta_{i,S}^{\text{DR}}$. Note that our approach can be easily adapted to other valuation schemes (i.e., Shapley value, Banzhaf value) since they are all based on marginal improvements (see Equ. (2)). More discussion on how our approach is related to current data valuation approaches is in Appendix A.

3.2. Model Deviation as Proxy for Marginal Improvement of DRGE

We define the model deviation of a data point z_i w.r.t. a dataset D_S for models in RKHS $\mathcal{H}$ as $\|f_S - f_{S\cup\{i\}}\|_{\mathcal{H}}$ where $f_S = \mathcal{A}(D_S)$ for a kernel-based algorithm $\mathcal{A}$. Then, the difference of the prediction results of $\forall x \in \mathcal{X}$ is bounded by the model deviation as $|f_S(x) - f_{S\cup\{i\}}(x)| \leq \|\Phi(x)\|_{\mathcal{H}}\|f_S - f_{S\cup\{i\}}\|_{\mathcal{H}}$. Hence, the model deviation essentially measures the difference in the prediction results of different functions, which is directly related to the performance of the model. The following theorem shows that model deviation can be a proxy for the marginal improvement of DRGE. We denote n as the size of $D_{S\cup\{i\}}$.

Theorem 3.3 (Bounded increase of DRGE for kernel-based algorithms)**.** Assume that $\|\Phi(x)\|_{\mathcal{H}} \leq M_0, \frac{q(x)}{p(x)} \leq M_1, \forall x \in \mathcal{X}, \forall Q \in \mathcal{Q}$; the loss function $\ell(\cdot,\cdot) \leq M_2$ and is L-Lipschitz continuous in its first argument for any fixed second argument. With probability at least $1-\delta$,

$$\begin{aligned}\Delta_{i,S}^{\text{DR}} &= R(f_S, \mathcal{Q}) - R(f_{S\cup\{i\}}, \mathcal{Q}) \\ &\leq \kappa_n \|f_S - f_{S\cup\{i\}}\|_{\mathcal{H}} + 2M_1M_2\sqrt{\ln(2/\delta)/2n}\end{aligned} \tag{4}$$

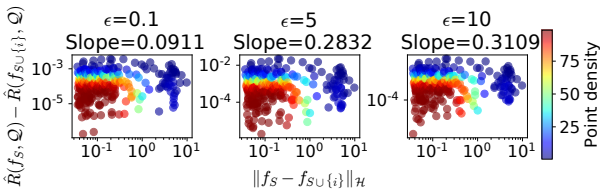

Figure 1. $\|f_S - f_{S\cup\{i\}}\|_{\mathcal{H}}$ and $\hat{R}(f_S, \mathcal{Q}) - \hat{R}(f_{S\cup\{i\}}, \mathcal{Q})$ for HOUSING. Slope is via linear regression of $\hat{R}(f_S, \mathcal{Q}) - \hat{R}(f_{S\cup\{i\}}, \mathcal{Q})$ on log-scaled $\|f_S - f_{S\cup\{i\}}\|_{\mathcal{H}}$.

where

$$\kappa_n = LM_0(1 + \sqrt{\varepsilon + \max\big((M_1-1)^2, 1\big)\sqrt{\ln(2/\delta)/2n}}).$$

The proof is in Appendix C. Intuitively, the generalization error of a model $f_{S\cup\{i\}}$ trained on the larger $S\cup\{i\}$ is expected to be lower than that of the model f_S trained on S. $R(f_S, \mathcal{Q}) - R(f_{S\cup\{i\}}, \mathcal{Q})$ indicates how much DRGE increases due to removing data point i from dataset $S\cup\{i\}$. Theorem 3.3 bounds this increase of DRGE, via the model deviation $\|f_S - f_{S\cup\{i\}}\|_{\mathcal{H}}$ multiplied by a constant κ_n. The intuition is that the less the model f_S deviates from the model $f_{S\cup\{i\}}$ (i.e., a better model), the prediction results by f_S are less different from those by $f_{S\cup\{i\}}$. Specifically, if ε in $\mathcal{Q}$ is small, κ_n will be small, so the effect of model deviation on the change in DRGE will be small. Intuitively, when Q is not allowed to shift further away from P, it cannot strategically put more density on the data point x where the prediction $f_S(x)$ is very different from the true label $f^*(x)$. Therefore, the DRGE cannot increase significantly in this case. As $n \to \infty$, the upper bound in Equ. (4) becomes tighter and eventually reduces to $\kappa_n\|f_S - f_{S\cup\{i\}}\|_{\mathcal{H}}$ where κ_n is the same for all data points. Consequently, Theorem 3.3 indicates that model deviation (i.e., $\|f_S - f_{S\cup\{i\}}\|_{\mathcal{H}}$) can be a proxy for the marginal improvement of DRGE and the proxy is potentially better when the dataset is larger. Further discussion on using model deviation as a proxy is in Appendix C.

We validate Theorem 3.3 on a real-world dataset (i.e., HOUSING in Sec. 5). For only the purpose of this empirical validation, we assume a very large dataset (the sampling distribution P) to implement DRGE $\hat{R}(f, \mathcal{Q})$ (details in Appendix A). Model deviation computation is in Sec. 3.3. Fig. 1 shows that, under varying ε, the change in DRGE (y-axis) decreases when model deviation (x-axis) decreases. The slope of the linear regression of $\hat{R}(f_S, \mathcal{Q}) - \hat{R}(f_{S\cup\{i\}}, \mathcal{Q})$ on log-scaled $\|f_S - f_{S\cup\{i\}}\|_{\mathcal{H}}$ decreases when ε decreases. This aligns with our analysis as κ_n in Equ. (4) will decrease as ε decreases. Consequently, $\hat{R}(f_S, \mathcal{Q}) - \hat{R}(f_{S\cup\{i\}}, \mathcal{Q})$ can increase less with the same amount of increase in $\|f_S - f_{S\cup\{i\}}\|_{\mathcal{H}}$ for a specific data point z_i.

3.3. Extending Model Deviation to NN

To resolve the difficulty of applying our approach to NN (i.e., the result in Theorem 3.3 does directly apply to DRGE of NN), we utilize the NTK theory to define model deviation for NN and show that it can be a proxy for marginal improvement of DRGE of NN. Recall that Θ is the NTK associated with the NN f (defined in Sec. 2). Denote f_S as an NN f trained on dataset D_S to convergence using gradient descent. Denote g_S as the minimizer of the kernel regression with NTK Θ trained on D_S. When the NN is wide enough, the output of f_S resembles that of g_S (Arora et al., 2019). We define the model deviation for NN as $\|g_S - g_{S\cup\{i\}}\|_{\mathcal{H}_\Theta}$. Denote λ_0 (λ_1) as the minimum eigenvalue of Θ_S ($\Theta_{S\cup\{i\}}$). We extend Theorem 3.3 to NN as follows,

Theorem 3.4 (Bounded increase of DRGE for NN)**.** Given the assumption in Theorem 3.3, further assume that $\|x\| = 1, \forall x \in \mathcal{X}$, $d_1 = d_2 \cdots = d_{L_{\text{NN}}} = m$ and $|f_0(x_i)| \leq \varepsilon_{\text{init}}, \forall\{x_i, y_i\} \in D_S$. Fix an $\varepsilon_k \leq \min\big(\text{poly}(1/L_{\text{NN}}, 1/(n-1), 1/\log(3/\delta), \lambda_0), \text{poly}(1/L_{\text{NN}}, 1/n, 1/\log(3/\delta), \lambda_1)\big)$. Set $m \geq \max(\text{poly}(1/\varepsilon_k, n-1, 1/\lambda_0), \text{poly}(1/\varepsilon_k, n, 1/\lambda_1))$. With probability at least $1-\delta$,

$$\Delta^{\text{DR}}_{i,S} = R(f_S, \mathcal{Q}) - R(f_{S\cup\{i\}}, \mathcal{Q})$$
$$\leq \kappa_n \|g_S - g_{S\cup\{i\}}\|_{\mathcal{H}_\Theta} + 2M_1M_2\sqrt{\ln(6/\delta)/2n} + \varepsilon_c$$

where $\varepsilon_c = 2L(\varepsilon_k + \varepsilon_{\text{init}})$ and

$$\kappa_n = LM_0(1 + \sqrt{\varepsilon + \max\big((M_1-1)^2, 1\big)\sqrt{\ln(6/\delta)/2n}}).$$

The randomness is over the random initialization of θ and the random sampling of $S\cup\{i\}$ from the distribution P. The proof is in Appendix C. The value of $\varepsilon_{\text{init}}$ is the magnitude of the model prediction of the dataset at initialization which is usually assumed to be small (Arora et al., 2019). The value of ε_k is smaller with a larger width m of the NN. When n is large, $2M_1M_2\sqrt{\ln(6/\delta)/2n}$ will be small. Therefore, Theorem 3.4 shows that the NTK-based model deviation is a good proxy of the marginal improvement of DRGE of corresponding NN when the NN is wide, its model output in initialization (i.e., $\varepsilon_{\text{init}}$) is small and the number n of data points is large. Importantly Theorem 3.4 extends Theorem 3.3 to NN, enabling model deviation to be applicable to settings adopting NN as the model. Note that our result is based on the previous result (Arora et al., 2019, Theorem 3.2), which can be extended to other NN architectures (e.g., convolutional neural network and ResNet), thus *not* limited to the fully connected NN.

Computing model deviation. For kernel ridge regression with kernel K, we utilize the closed-form solutions f_S and $f_{S\cup\{i\}}$ (see Sec. 2) to provide closed-form model deviations defined as $\mathcal{M}_i(K, D_S) \coloneqq \|f_S - f_{S\cup\{i\}}\|_{\mathcal{H}_K}$, computed as

$$\mathcal{M}_i(K, D_S)^2 = \boldsymbol{\alpha}^\top K_{S\cup\{i\}}\boldsymbol{\alpha} + \boldsymbol{\beta}^\top K_S\boldsymbol{\beta} - 2\boldsymbol{\beta}^\top K_{S,S\cup\{i\}}\boldsymbol{\alpha} \tag{5}$$

where $\boldsymbol{\alpha} \coloneqq (K_{S\cup\{i\}} + \lambda I)^{-1}\mathcal{Y}_{S\cup\{i\}}$ and $\boldsymbol{\beta} \coloneqq (K_S + \lambda I)^{-1}\mathcal{Y}_S$. As g_S is the kernel regression with NTK, a closed form is similarly available. The defined model deviation $\mathcal{M}_i(\Theta, D_S) \coloneqq \|g_S - g_{S\cup\{i\}}\|_{\mathcal{H}_\Theta}$ is computed as

$$\mathcal{M}_i(\Theta, D_S)^2 = \boldsymbol{\alpha}^\top \Theta_{S\cup\{i\}}\boldsymbol{\alpha} + \boldsymbol{\beta}^\top \Theta_S\boldsymbol{\beta} - 2\boldsymbol{\beta}^\top \Theta_{S,S\cup\{i\}}\boldsymbol{\alpha} \tag{6}$$

where $\boldsymbol{\alpha} = \Theta^{-1}_{S\cup\{i\}}\mathcal{Y}_{S\cup\{i\}}$ and $\boldsymbol{\beta} = \Theta_S^{-1}\mathcal{Y}_S$ and $\Theta_{S,S\cup\{i\}}$ is defined similarly as $K_{S,S\cup\{i\}}$. Importantly, Equ. (5) and Equ. (6) (explicitly derived in Appendix C) transform the computation of model deviation for both kernel ridge regression and NN into simple kernel evaluations.

However, a caveat of Equ. (6) is that it uses the theoretical NTK Θ that is difficult to compute due to the involved expectation and recursion (Lee et al., 2019), and lacks analytical form for most NN architecture. Fortunately, the theoretical NTK Θ is the limit of NTK at initialization Θ_0 (defined in Sec. 2) when the widths of the NN approach infinity and that the difference between Θ and Θ_0 is bounded for sufficiently wide NNs (Arora et al., 2019). Therefore, we can use the NTK at initialization (i.e., Θ_0) of a finite width NN to compute the model deviation, requiring only the gradient of model output w.r.t. the initial model parameter and with a bounded approximation error. Denote $\hat{\Theta}_S$ as the kernel matrix for NTK at initialization Θ_0 on the dataset D_S, we have the following:

Theorem 3.5 (Approximation error of using NTK at initialization)**.** For fixed $\varepsilon_k > 0$, assume that for each layer $\forall j \in \{1, \ldots, L_{\text{NN}}\}$, its width $d_j = \Omega(\frac{L^{14}_{\text{NN}}}{\varepsilon_k^4}\log(L_{\text{NN}}/\delta))$, each label $\forall y \in \mathcal{Y}, y \leq B$, and each input $\forall x \in \mathcal{X}, \|x\| \leq 1$. Then, $\forall\delta \in (0,1)$, with probability at least $1-\delta$,

$$|\mathcal{M}_i(\Theta, D_S) - \mathcal{M}_i(\Theta_0, D_S)| \leq C(L_{\text{NN}} + 1)\varepsilon_k$$

where $C = \frac{2n^3B^2}{M}\big(\frac{1}{\lambda_{\min}(\Theta_S)\lambda_{\min}(\Theta_{S\cup\{i\}})} + \frac{1}{\lambda_{\min}(\Theta_S)\lambda_{\min}(\hat{\Theta}_S)}\big)\big(\frac{1+\lambda_{\min}(\Theta_S)}{\lambda_{\min}(\hat{\Theta}_{S\cup\{i\}})}\big)$ and $\lambda_{\min}(\Theta_S)$ is the minimum eigenvalue of the matrix Θ_S and $M = \min(\mathcal{M}_i(\Theta, D_S), \mathcal{M}_i(\Theta_0, D_S))$.

The proof is in Appendix C. Theorem 3.5 bounds the approximation error from using Θ_0 (instead of Θ) for computing model deviation, via the product of a constant C, number of layers, and ε_k. Note that ε_k decreases as d_m increases, so for a sufficiently high d_m, the approximation error from using Θ_0 is (boundedly) small. Importantly, utilizing Θ_0 for computing model deviation implies a training-free approach since only the gradients of the model parameters at initialization are needed. This is desirable since most data valuation approaches need a large number of full training of NN (e.g.,

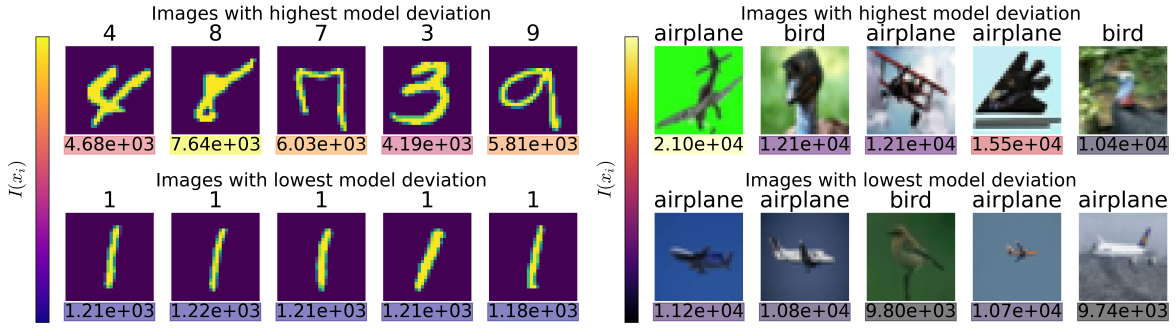

Figure 2. Images with highest (top) and lowest (bottom) **model deviation** for MNIST (left) and CIFAR-10 (right). Numbers below are $I(x_i)$, with higher values indicating greater dissimilarity within the class. Color bar shows relative $I(x_i)$ magnitude: lighter for higher, darker for lower.

$n+1$ for LOO) on different data subsets, which is computationally costly. In contrast, our training-free approach avoids such computational costs. We provide additional experiments to show that our approximation in Theorem 3.5 is both accurate and efficient in terms of running time in Appendix B. We also provide an extension of Theorem 3.5 that uses NTK at any training step t to approximate the model deviation in Appendix C.

4. Characteristics of Data Points That Lead to Low Model Deviations

We analyze the characteristics of data points in classification tasks with low model deviation under kernel ridge regression. A notion of uniqueness naturally arises and it consists of 1) *scarcity* (i.e., a small amount of data points with the same label) and 2) *dissimilarity* (i.e., different from other data points with the same label). Data points with low scarcity and/or dissimilarity lead to low model deviation.

Theorem 4.1 (Model deviation of kernel ridge regression)**.** Assume that $\|\Phi(x)\|_{\mathcal{H}} \leq M, \forall x \in \mathcal{X}$, $\max\{\|f_S\|_{\mathcal{H}}, \|f_{S\cup\{i\}}\|_{\mathcal{H}}\} = G$ and loss function ℓ is L-Lipschitz continuous. Assume that $\exists k$ s.t. $2n-1 \geq k \geq \frac{\|f_{S\cup\{i\}}\|_{\mathcal{H}}^2 - \|f_S\|_{\mathcal{H}}^2}{\|f_{S\cup\{i\}} - f_S\|_{\mathcal{H}}^2}$. Denote $Z_i := \{(x_j, y_j) : (x_j, y_j) \in D_S, y_j = y_i\}$ and $m = |Z_i|$. Then, $\forall \alpha \in [0,1]$,

$$\|f_S - f_{S\cup\{i\}}\|_{\mathcal{H}} \leq \frac{U + \sqrt{U^2 + 2\lambda(2n-\alpha-k)LGI(x_i)}}{\lambda(2n-\alpha-k)}$$

where $U = 2LM(n-m-1)$ and $I(x_i) = \sum_{(x_j,y_j)\in Z_i} \|\Phi(x_i) - \Phi(x_j)\|_{\mathcal{H}}$.

The proof is in Appendix C. We identify $U + \sqrt{U^2 + 2\lambda(2n-\alpha-k)LGI(x_i)}$ (i.e., numerator in r.h.s.) as the uniqueness of a data point z_i where a lower uniqueness implies a lower model deviation (Theorem 4.1). The uniqueness consists of two characteristics: 1) the *scarcity* and 2) the *dissimilarity* of the data point z_i. For scarcity, note that m is the number of data points with the same label as z_i. Hence, a low m can indicate that z_i is scarce. A lower scarcity (i.e., higher m) leads to a lower uniqueness and implies a lower model deviation of z_i. For dissimilarity, note that $I(x_i)$ is the sum of the distances of the data point z_i to every other data point z_j with the same label. A high $I(x_i)$ means that in the feature space (i.e., RKHS) the data point z_i is very far away from other data points with the same label. A lower dissimilarity (i.e., lower $I(x_i)$) leads to lower uniqueness and implies a lower model deviation of z_i.

Visualizing data points with different model deviations. Using Equ. (6) for model deviation on MNIST and CIFAR-10 (detailed in Appendix A), we perform a visual comparison: We construct a CIFAR-10 binary classification dataset with only bird and airplane classes (results on other classes of CIFAR-10 in Appendix B). We compute $I(x_i)$ for images. In Fig. 2, the images with high $\|f_S - f_{S\cup\{i\}}\|_{\mathcal{H}}$ tend to be more different from other images within the same class whilst images with low $\|f_S - f_{S\cup\{i\}}\|_{\mathcal{H}}$ usually are similar to other images in the same class (both visually and quantitatively with $I(x_i)$), confirming Theorem 4.1 that data points (i.e., images) with lower $I(x_i)$ have lower $\|f_S - f_{S\cup\{i\}}\|_{\mathcal{H}}$.

5. Experiments

We examine the effectiveness of using model deviation to value data points in data removal (Kwon & Zou, 2021; Xu et al., 2021; Just et al., 2023) and subset selection tasks (Liu et al., 2021; Song et al., 2021) since both are relevant to CML and data marketplace applications. For instance, in fixed-budget data purchases (Liu et al., 2021), subset selection aids buyers in buying a minimum DRGE data subset, which in turn ensures a maximized worst-case model performance and is thus crucial for potential buyers. Our code is available at https://github.com/xqlin98/Distributionally-Robust-Data-Valuation.

Table 1. DRGE (standard error) of the **kernel regression model (KR)** and **NN** trained on data subset (45% of the original dataset size) selected by different baselines. The lower the better.

Method	HOUSING	UBER	DIABETES	MNIST	CIFAR-10
			KR		
Random	2.530(5.6e-02)	392.563(1.0e+01)	0.926(7.6e-03)	1.976(2.2e-02)	2.726(3.7e-02)
LOO	**2.482**(3.5e-02)	372.124(8.5e+00)	**0.908**(5.7e-03)	2.075(3.6e-02)	2.479(1.7e-02)
TracIn	2.578(4.5e-02)	377.788(2.3e+01)	0.912(4.6e-03)	1.974(3.5e-02)	2.773(1.6e-02)
Influence	3.147(2.6e-01)	421.415(1.8e+01)	0.953(3.3e-02)	2.072(3.1e-02)	2.725(8.9e-03)
DAVINZ	2.623(8.4e-02)	402.609(2.0e+01)	0.919(1.0e-02)	2.400(1.5e-01)	2.988(1.7e-02)
CG	-	-	-	2.144(3.4e-02)	2.750(2.3e-02)
LAVA	-	-	-	2.031(2.7e-02)	3.006(3.1e-02)
Deviation	2.555(1.1e-01)	**361.810**(6.5e+00)	1.057(4.3e-02)	**1.893**(6.5e-02)	**2.470**(2.3e-02)
			NN		
Random	4.252(3.6e-02)	272.139(3.3e+00)	0.852(1.1e-02)	0.267(1.2e-02)	0.122(3.3e-04)
LOO	3.661(6.4e-02)	269.047(7.3e+00)	0.835(7.5e-03)	0.222(1.6e-02)	0.122(6.8e-04)
TracIn	4.193(6.8e-02)	311.670(9.9e+00)	0.853(4.2e-03)	0.264(1.3e-02)	0.123(3.7e-04)
Influence	4.674(1.2e-01)	279.930(7.1e+00)	0.839(2.4e-02)	0.320(9.7e-03)	0.126(5.8e-04)
DAVINZ	4.112(1.6e-01)	537.033(3.3e+01)	0.923(1.7e-02)	0.329(2.8e-02)	0.141(4.3e-03)
CG	-	-	-	**0.200**(1.5e-02)	0.123(3.7e-04)
LAVA	-	-	-	0.283(1.2e-02)	0.163(1.5e-03)
Deviation	**3.416**(5.0e-02)	**265.654**(9.3e+00)	**0.794**(6.1e-03)	0.224(1.4e-02)	**0.122**(6.4e-04)

Baselines. (a) **Random**, data values are sampled from the uniform distribution over $[0, 1]$. (b) **LOO**, leave-one-out score based on the validation performance. Since there is no known/fixed validation dataset available, we use the training dataset as the validation dataset. The same applies to other approaches that need validation datasets. (c) **Influence function** (Koh & Liang, 2017), an approach that approximates LOO with first-order extrapolation. (d) **TracIn** (Pruthi et al., 2020), an approach that computes the influence of a data point to the validation loss during training accumulatively. (e) **LAVA** (Just et al., 2023), a model-agnostic approach using a non-conventional class-wise Wasserstein distance to compute data values.[1] (f) **CG** (Nohyun et al., 2022), complexity-gap score which a training-free data valuation approach to quantify the influence of each data point to the generalization of a two-layer NN.[2] (g) **DAVINZ** (Wu et al., 2022), a training-free approach that uses an NTK-based utility function to approximate the generalization error. (h) **Deviation**, our approach.

Datasets. (a) HOUSING (Kaggle, 2017), California housing price prediction. (b) UBER (Kaggle, 2018), carpool ride price prediction. (c) DIABETES (Strack et al., 2014), diabetes patients' readmission prediction. (d) MNIST (LeCun et al., 1990). (e) CIFAR-10 (Krizhevsky, 2009).

Setting and evaluation metrics. There is a dataset of size n (e.g., from a seller) to be evaluated. The computed data values are used to perform data removal and subset selection. We consider two metrics: (a) DRGE, as the metric that the buyers are interested in (Sec. 2). To implement (i.e., empirically approximate this), we assume a very large sampling dataset with a size of $n_t \gg n$ as the empirical representation of the sampling distribution P and we obtain DRGE by solving an optimization problem (detailed in Appendix A). (b) Worst-case loss on multiple validation datasets, we perform a k-means clustering on the features of data points, using $k = 50$ to split the large sampling dataset into 50 validation datasets. We evaluate the worst validation loss among the 50 validation datasets, which simulates the case when there are a known number of buyers (e.g., 50) and we have access to their heterogeneous validation datasets. The use of k-means is to simulate the case that different buyers have different validation distributions. DRGE will serve as our primary evaluation metric. For all results, the average and standard error over 5 independent trials are reported.

Hyperparameters. We describe the hyperparameters for HOUSING and defer details for other datasets to Appendix A. The data sizes $n = 3000$ and $n_t = 15000$. For kernel regression, we use a radial basis function (RBF) kernel with a length scale of 2. For NN, we use a 3-layer multi-layer perceptron (MLP) for regression and a 2-layer convolutional neural network (CNN) followed by a fully connected layer for classification. The epoch number is 10 with a learning rate of 0.05 and batch size of 128.

5.1. Data Selection

For the task of data selection, we select 45% (results for 20% and 80% are provided in Appendix B) of data points with the highest data values and train models using these selected data points and evaluate the DRGE of the result-

[1] **LAVA** is designed for classification tasks.

[2] **CG** does not have implementation for regression.

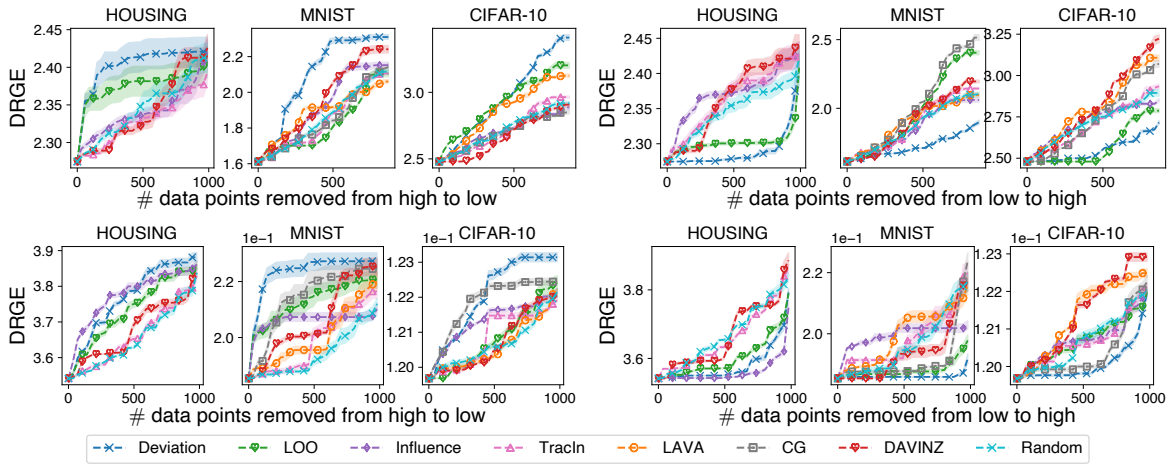

Figure 3. DRGE for kernel regression model (first row) and NN (second row) as data points are removed from the dataset. The first three columns are removing data points by the data values from high to low, the last three columns are removing data points by the data values from low to high.

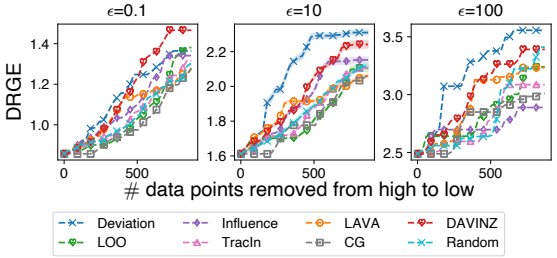

Figure 4. DRGE with different ε for kernel regression as data points are removed from MNIST by their data values from high to low.

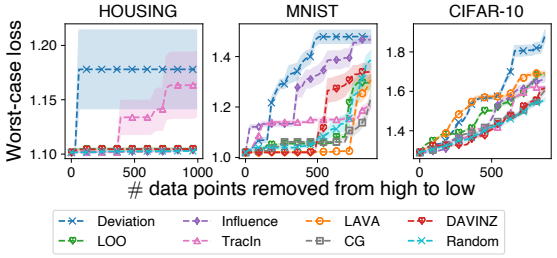

Figure 5. Worst-case performance for kernel regression as data points are removed by their data values from high to low.

ing model. Table 1 shows the results for kernel regression and NN. Remarkably, our approach achieves the lowest DRGE on UBER, MNIST, and CIFAR-10 in the kernel regression model and the lowest DRGE on all except MNIST in NN. LOO outperforms ours slightly on a few datasets, but markedly worse on the others. As NN-based approaches, Influence, TracIn, CG, and DAVINZ might not effectively reduce the DRGE of the kernel regression model. Consequently, these approaches are even worse compared to Random in the kernel regression model case. Surprisingly, these approaches also do not perform well in the NN case, including the algorithm-agnostic approach LAVA. One explanation is that these approaches basically approximate how much data points improve over some metric for one specific distribution. For example, DAVINZ is a proxy for the generalization error of NN on a specific validation distribution and LAVA is a proxy for the performance on one specific validation distribution. However, in our setting, since we are considering the worst-case (over a set of validation datasets) model performance, a data point that induces good performance on one specific validation dataset does not necessarily induce good worst-case model performance.

5.2. Data Removal

For data removal, each time we sample $m < n$ data points from the dataset (e.g., $m = 1000$ for HOUSING) and sequentially remove all m data points in a highest-value first (or lowest first) order. After each removal, we evaluate the DRGE of the model trained on the remaining dataset. Fig. 3 shows the results for the kernel regression (first row) and NN (second row). Additional results are in Appendix B. When removing data points from high to low data value (i.e., good data points are removed first), the faster the DRGE increases the better (i.e., the performance degrades faster). While removing from low to high, the slower the DRGE increases the better. Remarkably, our approach has the fastest increase of DRGE when removing from high to low while the slowest when removing from low to high in almost all scenarios, indicating that our approach outperforms other baselines. In Fig. 4, we vary ε in DRGE to see how different baselines perform (more choices of ε in Appendix B). Interestingly, when $\varepsilon \to 0$ ($\mathcal{Q}$ becomes a singleton and the DRGE reduces to generalization error), our approach is not significantly better. However, as ε increases to 100, our approach clearly outperforms others. Therefore, our approach performs better when the local validation datasets of buyers/parties are more heterogeneous (since a larger ε is

needed to make $\mathcal{Q}$ include all validation distributions). We also provide results using worst-case loss as the metric in Fig. 5, to simulate real-world applications as described in the evaluation metrics above. Our approach continues to outperform all baselines, since the worst-case model loss increases fastest when removing data points from high to low data value.

6. Related Work

Existing data valuation works assume the existence and availability of a known and fixed validation distribution (Ghorbani & Zou, 2019; Ghorbani et al., 2020; Kwon & Zou, 2021; Wang & Jia, 2023; Wu et al., 2024) and perform data valuation directly based on the validation performance. While (Ghorbani et al., 2020) considers data valuation when the dataset to be evaluated is not fixed, it does not address the challenge of unknown or unfixed validation distribution. For a more comprehensive survey, we refer the readers to (Sim et al., 2022). Other works (Wang et al., 2021; Xu et al., 2021; Wu et al., 2022; Just et al., 2023; Lin et al., 2023) find proxies to the validation performance. For example, (Xu et al., 2021) proposes a proxy that relaxes the availability of the validation distribution by removing access to the validation dataset. (Wu et al., 2022; Nohyun et al., 2022; Just et al., 2023) explore the training-free and model-agnostic proxy of generalization error respectively on specific validation distribution. However, these approaches still rely on a specific validation distribution and hence are not applicable without a known and fixed validation distribution. To address this, we propose a DRGE-based data valuation via the worst-case performance. Additional discussion on related work is in Appendix D.

7. Conclusion and Future Work

We exploit a *distributionally robust generalization error* to propose a novel data valuation framework that does not need a known/fixed validation distribution, yet provides a worst-case performance guarantee. We show that model deviation, a training-free proxy for the DRGE, can be used to perform data valuation to solve the intractability of DRGE. Interestingly, we identify a notion of uniqueness and show that data points with higher uniqueness will have higher model deviation, and thus be more valuable. We empirically demonstrate that our approach outperforms others in the task of data removal and data selection. Some future explorations present themselves: 1) Further relax the assumption on the buyers'/parties' local validation distributions so that our results can be applicable to a wider range of real-world scenarios; 2) Extend our approach to consider learning algorithms other than KR and NN. Investigating one or both is very useful in practice where different buyers can have very different local validation distributions and/or preferences for the learning algorithm.

Acknowledgements

This research/project is supported by the National Research Foundation Singapore and DSO National Laboratories under the AI Singapore Programme (AISG Award No: AISG2-RP-2020-018).

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none of which we feel must be specifically highlighted here.

References

Agarwal, A., Dahleh, M., and Sarkar, T. A marketplace for data: An algorithmic solution. In *Proc. EC, 2019*, 2019.

Amazon. Data exchange: PC/MS DOS word processing, spreadsheets, and databases, 4 2023. URL https://aws.amazon.com/data-exchange/. Accessed: 2023-04-23.

Arora, S., Du, S. S., Hu, W., Li, Z., Salakhutdinov, R. R., and Wang, R. On exact computation with an infinitely wide neural net. In *Proc. NeurIPS*, 2019.

Blanchet, J., Murthy, K., and Zhang, F. Optimal transport-based distributionally robust optimization: Structural properties and iterative schemes. *Mathematics of Operations Research*, 2022.

Cook, R. D. Detection of influential observation in linear regression. *Technometrics*, 19(1):15–18, 1977.

Dubey, P., Neyman, A., and Weber, R. J. Value theory without efficiency. *Mathematics of Operations Research*, 6(1):122–128, 1981.

Ghorbani, A. and Zou, J. Data Shapley: Equitable valuation of data for machine learning. In *Proc. ICML*, 2019.

Ghorbani, A., Kim, M. P., and Zou, J. A distributional framework for data valuation. In *Proc. ICML*, 2020.

Ghorbani, A., Zou, J., and Esteva, A. Data Shapley valuation for efficient batch active learning. *arXiv preprint arXiv:2104.08312*, 2021.

Gotoh, J.-y., Kim, M. J., and Lim, A. E. Robust empirical optimization is almost the same as mean–variance optimization. *Operations research letters*, 46(4):448–452, 2018.

Hemachandra, A., Dai, Z., Singh, J., Ng, S.-K., and Low, B. K. H. Training-free neural active learning with initialization-robustness guarantees. In *Proc. ICML*, 2023.

Hu, W., Niu, G., Sato, I., and Sugiyama, M. Does distributionally robust supervised learning give robust classifiers? In *Proc. ICML*, 2018.

Jia, R., Dao, D., Wang, B., Hubis, F. A., Hynes, N., Gürel, N. M., Li, B., Zhang, C., Song, D., and Spanos, C. J. Towards efficient data valuation based on the Shapley value. In *Proc. AISTATS*, 2019.

Just, H. A., Kang, F., Wang, T., Zeng, Y., Ko, M., Jin, M., and Jia, R. LAVA: Data valuation without pre-specified learning algorithms. In *Proc. ICLR*, 2023.

Kaggle. California housing prices, 2017. URL https://www.kaggle.com/datasets/camnugent/california-housing-prices. Dataset.

Kaggle. Uber and Lyft dataset, 2018. URL https://www.kaggle.com/datasets/brllrb/uber-and-lyft-dataset-boston-ma. Dataset.

Koh, P. W. and Liang, P. Understanding black-box predictions via influence functions. In *Proc. ICML*, 2017.

Krizhevsky, A. Learning multiple layers of features from tiny images. Master's thesis, Department of Computer Science, University of Toronto, 2009.

Kwon, Y. and Zou, J. Beta Shapley: a unified and noise-reduced data valuation framework for machine learning. In *Proc. AISTATS*, 2021.

Lau, G. K. R., Hemachandra, A., Ng, S.-K., and Low, B. K. H. PINNACLE: PINN Adaptive ColLocation and Experimental points selection. In *Proc. ICLR*, 2024.

LeCun, Y., Boser, B., Denker, J., Henderson, D., Howard, R., Hubbard, W., and Jackel, L. Handwritten digit recognition with a back-propagation network. In *Proc. NeurIPS*, 1990.

Lee, J., Xiao, L., Schoenholz, S., Bahri, Y., Novak, R., Sohl-Dickstein, J., and Pennington, J. Wide neural networks of any depth evolve as linear models under gradient descent. In *Proc. NeurIPS*, 2019.

Li, T., Hu, S., Beirami, A., and Smith, V. Ditto: Fair and robust federated learning through personalization. In *Proc. ICML*, 2021.

Lin, X., Xu, X., Ng, S.-K., Foo, C.-S., and Low, B. K. H. Fair yet asymptotically equal collaborative learning. In *Proc. ICML*, 2023.

Lin, X., Xu, X., Wu, Z., Sim, R. H. L., Ng, S.-K., Foo, C.-S., Jaillet, P., Hoang, T. N., and Low, B. K. H. Fairness in federated learning. In *Federated Learning*, pp. 143–160. Elsevier, 2024.

Liu, J., Lou, J., Liu, J., Xiong, L., Pei, J., and Sun, J. Dealer: an end-to-end model marketplace with differential privacy. 2021.

McMahan, B., Moore, E., Ramage, D., Hampson, S., and y Arcas, B. A. Communication-efficient learning of deep networks from decentralized data. In *Proc. AISTATS*, 2017.

Namkoong, H. and Duchi, J. C. Variance-based regularization with convex objectives. In *Proc. NeurIPS*, 2017.

Nohyun, K., Choi, H., and Chung, H. W. Data valuation without training of a model. In *Proc. ICLR*, 2022.

Pruthi, G., Liu, F., Kale, S., and Sundararajan, M. Estimating training data influence by tracing gradient descent. In *Proc. NeurIPS*, 2020.

Rahimian, H. and Mehrotra, S. Distributionally robust optimization: A review. *arXiv preprint arXiv:1908.05659*, 2019.

Shapley, L. S. A value for n-person games. *Annals of Mathematics Studies*, 1953.

Sim, R. H. L., Zhang, Y., Chan, M. C., and Low, B. K. H. Collaborative machine learning with incentive-aware model rewards. In *Proc. ICML*, 2020.

Sim, R. H. L., Xu, X., and Low, B. K. H. Data valuation in machine learning: "ingredients", strategies, and open challenges. In *Proc. IJCAI*, 2022.

Song, Q., Cao, J., Sun, K., Li, Q., and Xu, K. Try before you buy: Privacy-preserving data evaluation on cloud-based machine learning data marketplace. In *Proc. ACSAC*, 2021.

Staib, M. and Jegelka, S. Distributionally robust optimization and generalization in kernel methods. In *Proc. NeurIPS*, 2019.

Strack, B., DeShazo, J. P., Gennings, C., Olmo, J. L., Ventura, S., Cios, K. J., and Clore, J. N. Impact of HbA1c measurement on hospital readmission rates: analysis of 70,000 clinical database patient records. *BioMed research international*, 2014, 2014.

Tu, X., Zhu, K., Luong, N. C., Niyato, D., Zhang, Y., and Li, J. Incentive mechanisms for federated learning: From economic and game theoretic perspective. *IEEE Transactions on Cognitive Communications and Networking*, 2022.

Walker, R. 117Strategies for Monetizing Big Data. In *From Big Data to Big Profits: Success with Data and Analytics*. Oxford University Press, 08 2015. ISBN 9780199378326. doi: 10.1093/acprof:oso/9780199378326.003.0007.

Wang, J. T. and Jia, R. Data Banzhaf: A robust data valuation framework for machine learning. In *Proc. AISTATS*, 2023.

Wang, T., Rausch, J., Zhang, C., Jia, R., and Song, D. A principled approach to data valuation for federated learning. *Lecture Notes in Computer Science*, 12500:153–167, 2020.

Wang, T., Yang, Y., and Jia, R. Improving cooperative game theory-based data valuation via data utility learning. *arXiv preprint arXiv:2107.06336*, 2021.

Wu, Z., Shu, Y., and Low, B. K. H. DAVINZ: Data valuation using deep neural networks at initialization. In *Proc. ICML*, 2022.

Wu, Z., Xu, X., Sim, R. H. L., Shu, Y., Lin, X., Agussurja, L., Dai, Z., Ng, S.-K., Foo, C.-S., Jaillet, P., Hoang, T. N., and Low, B. K. H. Data valuation in federated learning. In *Federated Learning*, pp. 281–296. Elsevier, 2024.

Xu, X., Wu, Z., Foo, C. S., and Low, B. K. H. Validation free and replication robust volume-based data valuation. In *Proc. NeurIPS*, 2021.

Yu, H. and Zhang, M. Data pricing strategy based on data quality. *Computers & Industrial Engineering*, 112:1–10, 2017.

Zhang, X., Chan, F. T., Yan, C., and Bose, I. Towards risk-aware artificial intelligence and machine learning systems: An overview. *Decision Support Systems*, pp. 113800, 2022.

A. Additional Details on Experiment Settings

A.1. Dataset License and Computational Resource

Licence. HOUSING (Kaggle, 2017): CC0 1.0 Universal (CC0 1.0) Public Domain Dedication; UBER (Kaggle, 2018): CC0 1.0 Universal (CC0 1.0) Public Domain Dedication; DIABETES (Strack et al., 2014): Open Data Commons Open Database License (ODbL) v1.0; MNIST (LeCun et al., 1990): Attribution-Share Alike 3.0 License; CIFAR-10 (Krizhevsky, 2009): MIT License.

Computational resource. All the experiments have been run on a server with Intel(R) Xeon(R) Gold 6226R CPU @ 2.90GHz processor, 256GB RAM, and 4 NVIDIA GeForce RTX 3080s.

A.2. Discussion on the Definition of $\mathcal{Q}$

Advantages of choosing χ^2-divergence in defining $\mathcal{Q}$. We choose χ^2-divergence to define $\mathcal{Q}$ due to its simple form which is amenable to our theoretical result and its applicability to the data marketplace. χ^2-divergence is commonly adopted in the distributionally robust optimization literature (Namkoong & Duchi, 2017; Gotoh et al., 2018) due to its simplicity and rich theoretical results. Specifically, the definition of χ^2-divergence is amenable to the derivation of our Theorem 3.3 and 3.4. Additionally, the simplicity of its form makes it easier to compute than other distance measures (e.g., optimal transport). On the other hand, the intuition behind χ^2-divergence is applicable to the data marketplace applications since it effectively measures the discrepancy between the densities of different subregions in the two compared distributions. To elaborate, in practice, the buyers' distribution is usually confined to some subregions of the overall distribution. For example, the patient's data varies across hospitals due to different demographics (e.g., different densities in race due to different geographic locations or different densities in gender due to different specialties of the hospitals). The χ^2-divergence is particularly useful here because it measures this discrepancy and therefore results in a meaningful set of distributions $\mathcal{Q}$.

Reasons of choosing $\chi^2(Q, P)$ instead of $\chi^2(P, Q)$ to define $\mathcal{Q}$. Recall that we define $\mathcal{Q}$ using $\chi^2(Q, P)$, which requires the support of P to be a superset of the support of Q. In the application of data marketplace, it implies that the buyer's validation distribution may be restricted to a region of the support while the seller's distribution is more extensive (in terms of support), which is the main reason that the buyer wants to buy data from the seller. In contrast, the opposite direction implies that the buyer's data is actually more extensive (in terms of support) than that of the seller, and then the buyer may not be interested in buying the data from the seller.

Comparison to other distributional distance measures. There are quite a few distributional distance measures in the current literature (Staib & Jegelka, 2019; Blanchet et al., 2022), among which maximum mean discrepancy (MMD) and Wasserstein distance (i.e., optimal transport) are commonly considered. Specifically, MMD measures the distance of two distributions by the square of the distance of their embedding in the RKHS associated with a specific kernel. Therefore, it is dependent on the choice of kernel and thus a poor choice of kernel might result in a poor measure of distributional distance. Additionally, the approximation of worst-case performance (provided in (Staib & Jegelka, 2019)) based on the MMD uncertainty set is only applicable to kernel-based algorithms. In contrast, our model deviation can be applied to both kernel-based algorithms and neural networks. As for the Wasserstein distance (i.e., optimal transport), it can hardly model the complex real-world dataset due to the use of simple metrics (e.g., the Euclidean or Mahalanobis distance). Specifically, since the Wasserstein distance does not impose constraints on the structure and properties of the data, it might result in a meaningless measure of the distributional distance. For example, simple metrics (e.g., Euclidean distance) will not be able to measure the distance between two images in an image classification dataset (in the sense that two images with the same semantics will have very large Euclidean distance), and hence the uncertainty set using this metric might not be as meaningful. Additionally, Wasserstein distance is computationally difficult since it requires solving a non-trivial optimization (i.e., optimal transport problem). In contrast, the simple formation of χ^2-divergence makes it much easier to compute without solving the optimization problem.

Difference of our definition of $\mathcal{Q}$ from the other literature that use φ-divergence. χ^2-divergence is one specific kind of φ-divergence. There is some literature in distributionally robust optimization that defines the uncertainty set $\mathcal{Q}$ with φ-divergence (Namkoong & Duchi, 2017; Gotoh et al., 2018). However, (Staib & Jegelka, 2019) points out that φ-divergence is usually defined using the empirical distribution $\hat{P}$ with limited support, which cannot handle out-of-sample cases. To elaborate, if we define $\mathcal{Q}$ using φ-divergence based on the empirical distribution $\hat{P}$, all the distributions in $\mathcal{Q}$ will

have the same support as $\hat{P}$. Consequently, if there is one buyer whose validation distribution has a support that is not a subset of the support of $\hat{P}$, the worst-case model performance on $\mathcal{Q}$ cannot say anything about the model performance on the validation distribution of that buyer. In contrast, we define $\mathcal{Q}$ by using the distribution P (instead of empirical distribution $\hat{P}$) where the training data are sampled from and thus mitigate this problem.

Applicability of our definition of $\mathcal{Q}$ to real-world applications. For distributionally robust generalization error discussed in Sec.3 using our definition of $\mathcal{Q}$ to guarantee the worst model performance in buyers'/parties' local validation distribution, we need to assume that the parties/buyers' validation distributions are within the defined uncertainty set which might not be able to handle the case where some parties/buyers' validation distributions are not within the set. This can be addressed by carefully choosing the reference distribution P and ε in the definition of $\mathcal{Q}$. Nonetheless, Fig.5 shows that our approach still performs significantly well when this assumption is not fulfilled.

A.3. Discussion on the Difference Between Our Approach and Current Data Valuation Approaches (e.g., LOO and Data Shapley which Uses Shapley Value)

In summary, our model deviation approximates the data values with DRGE as the utility function and leave-one-out (LOO) as the way to attribute the utility to each individual data point.

To elaborate, current data valuation approaches are heavily dependent on two ingredients: 1) The utility function U to measure the contribution of a set of data points and 2) The valuation function defined in Equ. (2) that attributes the utility to each data point (e.g., Shapley value (Shapley, 1953), LOO (Cook, 1977), semivalue (Dubey et al., 1981)). Current work usually assumes access to a validation dataset and thus the utility function is the validation performance. Existing work adopts Shapley value as a valuation function (e.g., data Shapley (Ghorbani & Zou, 2019) and beta Shapley (Kwon & Zou, 2021)) with validation performance as the utility function. Differently, we propose DRGE as a new utility function, and it can be combined with different valuation functions (e.g., LOO or Shapley value, we use LOO in our paper). In particular, our proposed model deviation, as the proxy for marginal contributions, can be applied well to any valuation functions mentioned above, since all require marginal contributions. As a special case, LOO requires one single marginal contribution and thus can be approximated easily by our model deviation. Therefore, our approach can be combined with other valuation functions (e.g., Shapley value or semivalue) to get additional properties according to the needs of the application scenarios.

A.4. Approximation of DRGE for the Evaluation Metric

Approximation of DRGE To approximate the distributionally robust generalization error, we assume that there exists a very large dataset with a size denoted as n_t, which is much larger (e.g., more than $10,000$) than the size of the dataset to be evaluated, denoted as n (e.g., around $1,000 - 3,000$). It is i.i.d. sampled from the sampling distribution P (i.e., the distribution where the dataset to be evaluated is sampled from). We define the empirical data distribution as $\hat{P}$ which puts equal density on each data point in the large dataset. We assume that we know the ε to define $\mathcal{Q}$ such that all possible validation distributions of the buyers/parties are within the $\mathcal{Q}$. Therefore, we are able to estimate DRGE (in Equ. (3)) for the trained model f by solving the following quadratically constrained linear programming (QCLP):

$$
\begin{aligned}
\hat{R}(f, \mathcal{Q}) \coloneqq \underset{p}{\text{maximize}} \quad & \frac{1}{n_t} \sum_{i=1}^{n_t} p_i \ell(f(x_i), y_i) && \text{(Objective)} \\
\text{subject to} \quad & p_i \geq 0, \quad \forall i \in \{1, ..., n_t\} && \text{(C1)} \\
& \sum_{i=1}^{n_t} p_i = 1 && \text{(C2)} \\
& \frac{1}{n_t} \| n_t p - 1 \|_2^2 \leq \varepsilon && \text{(C3)}
\end{aligned}
$$

The constraints C1 and C2 are to ensure that p is a valid distribution. The constraint C3 is to ensure that p is within the $\mathcal{Q}$ defined with $\hat{P}$. The objective is the loss of the model f on the distribution p. Therefore, we use the maximized objective $\hat{R}(f, \mathcal{Q})$ of the QCLP to approximate the DRGE $R(f, \mathcal{Q})$. Of note, the assumptions we make here, namely the availability of the large dataset and the knowledge of ε, are *not* the assumptions for our approach, but the assumptions for obtaining a viable evaluation metric.

Difficulties of directly using the DRGE approximation to perform data valuation There are two difficulties in using the above-mentioned approximation approach to perform data valuation, namely: 1) the availability of a large dataset that contains sufficient support and information on ε, and 2) computational efficiency. Specifically, in real-life applications, the large dataset is usually not available since it needs to be large enough to have large support and thus is able to approximate DRGE accurately. Otherwise, the out-of-sample problem (discussed in Appendix A.2) is more likely to arise due to the small support. However, for example, it is impractical to assume the availability of a large dataset that includes the support of all validation distributions of the buyers/parties due to the potentially high heterogeneity among these distributions in real-life applications. It is also unclear how to find the value of ε such that all possible validation distributions are within $\mathcal{Q}$ to provide the worst-case performance guarantee. As for the computational complexity, data valuation approaches typically require a large number of complete model trainings on different data subsets (e.g., $n+1$ number of trainings for the LOO approach as we discussed in Sec. 3). Consequently, the computational costs can be substantial for these approaches, especially when NN is used. Due to these difficulties, the approximated DRGE is only used as an evaluation metric under desirable conditions to evaluate the performance of different baselines.

A.5. Experiment Settings for Visualization of Data Points with Different Model Deviations in Sec. 4.

We use NTK at initialization Θ_0 to compute the model deviation $\|f_S - f_{S\cup\{i\}}\|_{\mathcal{H}}$ (i.e., $\mathcal{H} = \mathcal{H}_{\Theta_0}$). To compute the NTK at initialization, we use the CNN model with two convolutional layers followed by a fully connected layer for both MNIST and CIFAR-10. We randomly select 2000 data points from the training dataset to evaluate their model deviation and plot the images with the highest and lowest model deviations. The computation of $I(x_i)$ can be converted to kernel evaluation as follows:

$$\begin{aligned} I(x_i) &= \sum_{(x_j,y_j)\in Z_i} \|\Phi(x_i) - \Phi(x_j)\|_{\mathcal{H}_{\Theta_0}} \\ &= \sum_{(x_j,y_j)\in Z_i} \sqrt{\langle \Phi(x_i) - \Phi(x_j), \Phi(x_i) - \Phi(x_j)\rangle_{\mathcal{H}_{\Theta_0}}} \\ &= \sum_{(x_j,y_j)\in Z_i} \sqrt{\Theta_0(x_i,x_i) + \Theta_0(x_j,x_j) - 2\Theta_0(x_i,x_j)}\,. \end{aligned}$$

A.6. Hyperparameters for the Experiments in Sec. 5

Table 2. Hyperparameters for kernel regression model

Dataset	n	n_t	m	Length scale	λ
HOUSING	3000	15000	1000	2	1e-2
UBER	1500	30000	1000	3	1e-2
DIABETES	4000	20000	2000	10	1e-2
MNIST	1000	10000	900	50	0
CIFAR-10	1000	10000	900	600	5e-4

Tab. 2 shows the hyper-parameters for the kernel regression model (specifically, kernel ridge regression with the regularization parameter λ) for each dataset. Note that the numbers of data points (i.e., n, n_t, and m) are chosen based on the size of the original dataset. The length scale and λ are set based on the performance of the kernel regression model. Specifically, the length scale is set by referring to the median value of the pairwise Euclidean distance of the data points in the dataset.

Tab. 3 shows the hyperparameters for the NN for each dataset. The learning rate and the number of epochs are set based on the performance of NN for each dataset.

B. Additional Experiment Results

B.1. Additional Experiment Results for Validating the Efficiency of Approximation with NTK at Initialization

To provide a more detailed empirical verification on the approximation accuracy and approximation efficiency of Theorem 3.5, we perform additional experiments. Specifically, we compute the model deviation using NTK at initialization (i.e.,

Table 3. Hyperparameters for NN

Dataset	n	n_t	m	Learning rate	Epoch	NN architecture
HOUSING	3000	15000	1000	5e-3	10	Linear(input dimensions, 128)-ReLU Linear(128, 128)-ReLU Linear(128, 1)
UBER	1000	30000	900	1e-3	50	Linear(input dimensions, 128)-ReLU Linear(128, 128)-ReLU Linear(128, 1)
DIABETES	4000	20000	2000	5e-3	10	Linear(input dimensions, 128)-ReLU Linear(128, 128)-ReLU Linear(128, 1)
MNIST	2000	10000	1000	1e-2	50	Conv2d(1, 16, 5, 1, 2)-ReLU-MP(2) Conv2d(16, 32, 5, 1, 2)-ReLU-MP(2) Linear(32 * 7 * 7, 10)
CIFAR-10	2000	10000	1000	5e-3	20	Conv2d(3, 16, 5, 1, 2)-ReLU-MP(2) Conv2d(16, 32, 5, 1, 2)-ReLU-MP(2) Linear(32 * 7 * 7, 10)

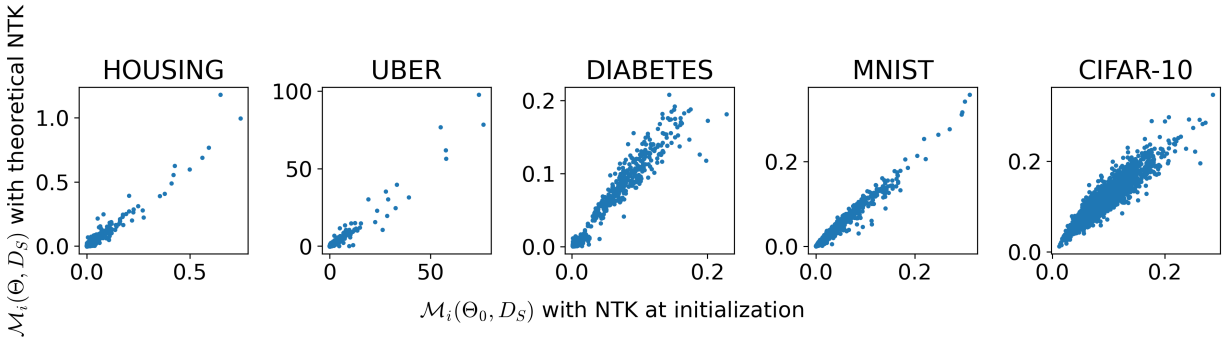

Figure 6. Pearson correlation between $\mathcal{M}_i(\Theta, D_S)$ and $\mathcal{M}_i(\Theta_0, D_S)$ on different datasets.

$\mathcal{M}_i(\Theta_0, D_S)$) and model deviation with theoretical NTK (i.e., $\mathcal{M}_i(\Theta, D_S)$) for different datasets with different models. Fig. 6 shows that the points lie in the diagonal of the figure which means that the approximated model deviations are almost perfectly correlated with the ground true model deviations. This result shows that our approximation is accurate across different datasets and models. Moreover, to further validate that approximation accuracy improves when the width of the NN increases, Fig. 7 shows the Pearson correlation between the approximated model deviation and the ground true model deviation. As the width of the NN increases, the correlation increases, and hence the accuracy of approximation increases, which coincides with our theoretical analysis in Theorem 3.5. Table 4 shows the running time of computing model deviation data values for different datasets, the time used for model deviation computation is low since no model training is required.

In summary, our model deviation approximation in Theorem 3.5 is accurate and efficient in terms of running time.

B.2. Additional Experiment Results for Visualization of Data Points with Different Model Deviations/LOO Scores

In Sec. 4, we visualize the images with different model deviations for MNIST and CIFAR-10 in Fig. 2. For comparison, we provide images with different LOO scores in Fig. 8. We can observe that images with high LOO scores are not dissimilar to other data points from the same class, both visually and quantitatively, as indicated by $I(x_i)$. This is in contrast to Fig. 2,

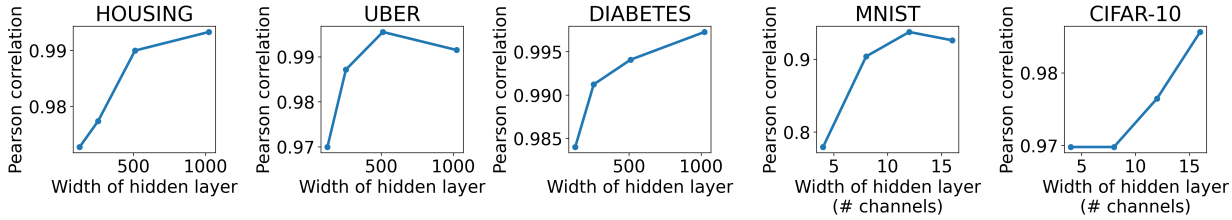

Figure 7. Pearson correlation between $\mathcal{M}_i(\Theta, D_S)$ and $\mathcal{M}_i(\Theta_0, D_S)$ with different widths of the hidden layer.

Table 4. Running time for computing model deviation data values for different datasets

Dataset	Running time
HOUSING	17.18 mins
UBER	0.65 mins
DIABETES	39.56 mins
MNIST	13.18 mins
CIFAR-10	12.76 mins

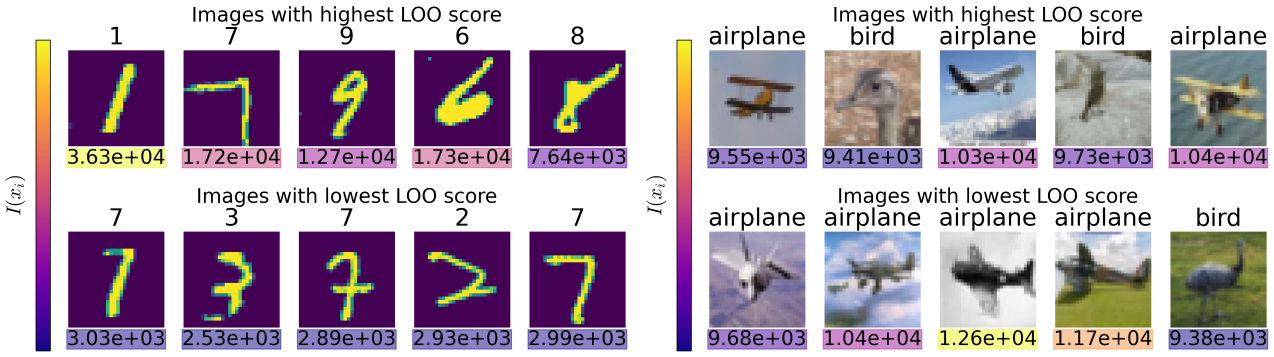

Figure 8. Images with highest (top) and lowest (bottom) **LOO scores** for MNIST (left) and CIFAR-10 (right). Numbers below represent $I(x_i)$, with higher values indicating greater dissimilarity within the class. Color bar shows relative $I(x_i)$ magnitude: lighter for higher, darker for lower.

where images with high (low) model deviations are clearly dissimilar (similar) to other data points from the same class.

In the experiments for Fig. 2 and Fig. 8, we construct the CIFAR-10 binary classification dataset using only two classes (i.e., bird and airplane). Here, we additionally conduct experiments on the CIFAR-10 binary classification dataset with different classes. Fig. 9 shows the images with high/low model deviations/LOO scores for the CIFAR-10 binary classification dataset with only cat and dog classes. Fig. 10 shows the result for deer and horses classes. Fig. 11 shows the result for car and truck classes. We can draw the same conclusion as we get from Sec. 4 that images with high (low) model deviations tend to be dissimilar (similar) to other images from the same class both visually and quantitatively with $I(x_i)$, which is more significant when compared with the result of LOO scores.

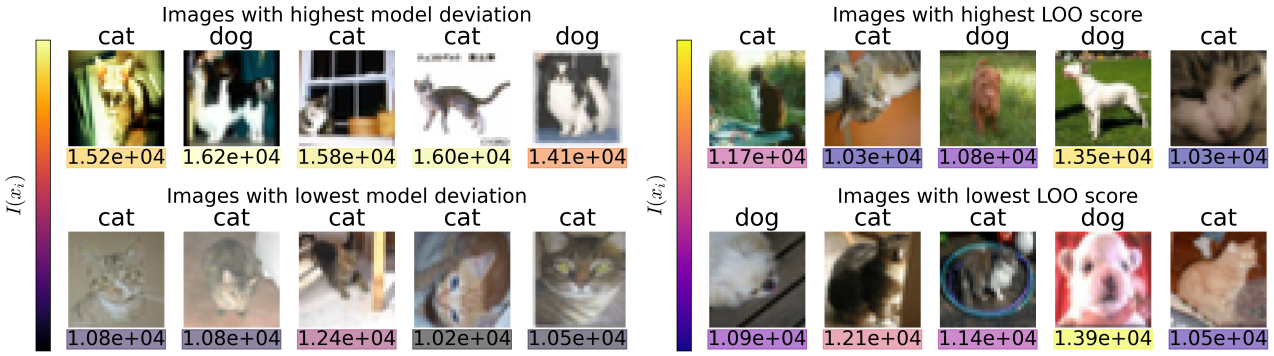

Figure 9. Images with highest (top) and lowest (bottom) model deviations (left) and LOO scores (right) for binary CIFAR-10 dataset with **only cat and dog classes**. Numbers below represent $I(x_i)$, with higher values indicating greater dissimilarity within the class. Color bar shows relative $I(x_i)$ magnitude: lighter for higher, darker for lower.

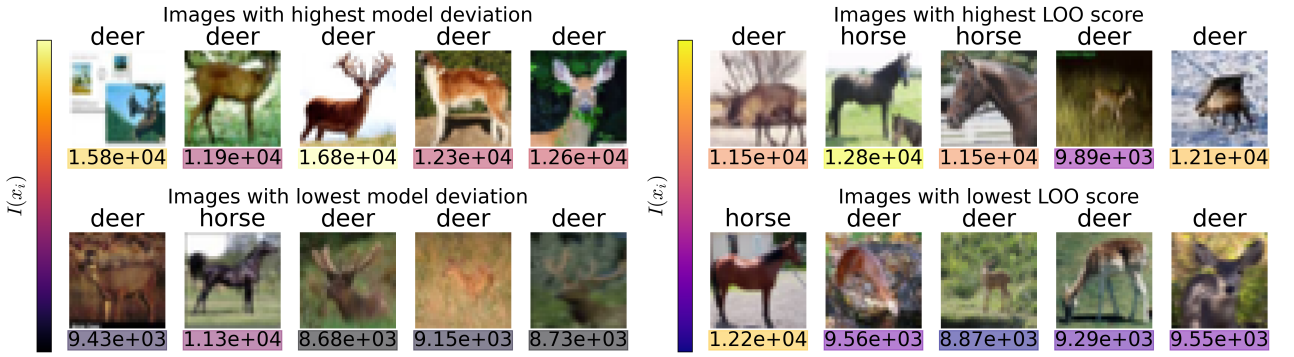

Figure 10. Images with highest (top) and lowest (bottom) model deviations (left) and LOO scores (right) for binary CIFAR-10 dataset with **only deer and horse classes**. Numbers below represent $I(x_i)$, with higher values indicating greater dissimilarity within the class. Color bar shows relative $I(x_i)$ magnitude: lighter for higher, darker for lower.

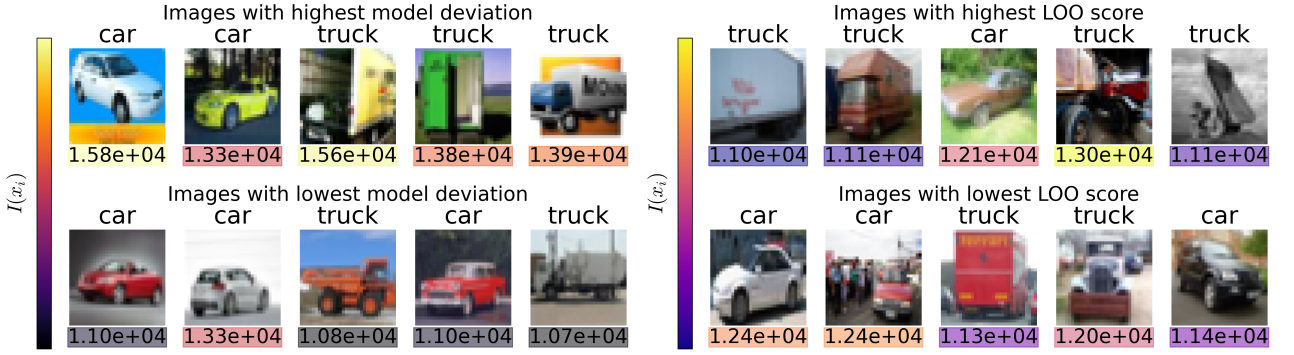

Figure 11. Images with highest (top) and lowest (bottom) model deviations (left) and LOO scores (right) for binary CIFAR-10 dataset with **only car and truck classes**. Numbers below represent $I(x_i)$, with higher values indicating greater dissimilarity within the class. Color bar shows relative $I(x_i)$ magnitude: lighter for higher, darker for lower.

B.3. Additional Experiment Results for Subset Selection

In Sec. 5, we perform the data selection task which selects 45% of data points from the original dataset using different baselines. We additionally provide the data selection results of selecting 80% (Tab. 5) and 20% (Tab. 6) of data points from the original dataset. Our approach still performs better than other baselines on most datasets.

B.4. Additional Experiment Results for Data Removal

In Sec. 5, we only provide data removal results for selected datasets (i.e., HOUSING, MNIST, and CIFAR-10) in Fig. 3 due to the limited space. Here, we provide full results for all the datasets introduced in Sec. 5 for both kernel regression model (Fig. 12) and NN (Fig. 13). In Fig. 12 and Fig. 13, our approach performs the best on almost all datasets for both kernel regression model and NN.

In Sec. 5, we provide results on how the performance of different baselines change when ε changes among $\{1, 10, 100\}$ in Fig. 4. Here, we provide the results of more choices of ε (i.e., $\varepsilon \in \{0.01, 0.1, 1, 10, 100\}$) in Fig. 14. We can draw the same conclusion as we get in Sec. 5 that our approach performs significantly better than other baselines when the degree of heterogeneity among the local validation distributions of the buyers/parties is large (e.g., $\varepsilon = 10$ or $\varepsilon = 100$).

We provide further analysis of the data removal result by examining the pointwise loss of the large dataset (used to approximate DRGE) of the model trained on the dataset after different numbers of data points are removed using model deviation and LOO score. We use the kernel regression model on the MNIST dataset for the experiment. Fig. 16 shows the histogram of the pointwise loss after removing data points with different approaches. When removing data points from high to low data values (first row), the density of data points with high loss increases more significantly when using model

Table 5. DRGE (standard error) of the **kernel regression model (KR)** and **NN** trained on data subset (80% of the original dataset size) selected by different baselines. The lower the better.

	Method	HOUSING	UBER	DIABETES	MNIST	CIFAR-10
KR	Random	2.443(4.9e-02)	383.537(1.9e+01)	0.899(6.9e-03)	1.703(2.5e-02)	2.554(2.0e-02)
	LOO	2.444(3.3e-02)	366.667(7.7e+00)	0.885(4.8e-03)	**1.670**(2.7e-02)	**2.395**(1.5e-02)
	TracIn	2.374(2.2e-02)	354.212(1.3e+01)	0.892(3.4e-03)	1.696(2.7e-02)	2.577(1.5e-02)
	Influence	2.630(9.4e-02)	388.952(1.7e+01)	0.914(1.9e-03)	1.791(3.9e-02)	2.597(1.4e-02)
	DAVINZ	2.444(3.0e-02)	358.615(9.6e+00)	0.895(5.6e-03)	1.820(2.2e-02)	2.624(2.9e-02)
	LAVA	-	-	-	1.743(2.3e-02)	2.615(1.1e-02)
	CG	-	-	-	2.787(9.8e-02)	3.191(8.5e-02)
	Deviation	**2.349**(1.3e-02)	**352.917**(6.3e+00)	**0.876**(4.6e-03)	1.719(4.1e-02)	2.460(1.7e-02)
NN	Random	3.646(3.4e-02)	268.812(4.8e+00)	0.850(6.3e-03)	0.235(1.4e-02)	0.119(1.7e-04)
	LOO	3.501(3.8e-02)	**261.059**(5.6e+00)	0.845(4.9e-03)	0.228(1.6e-02)	0.120(1.5e-04)
	TracIn	3.625(3.9e-02)	272.611(4.3e+00)	0.847(4.7e-03)	0.230(1.3e-02)	0.119(2.0e-04)
	Influence	4.095(5.5e-02)	273.224(7.4e+00)	0.860(1.8e-02)	0.284(1.4e-02)	0.121(3.6e-04)
	DAVINZ	3.655(4.2e-02)	288.686(1.2e+01)	0.898(3.5e-03)	0.250(1.5e-02)	0.122(7.8e-04)
	LAVA	-	-	-	0.248(2.0e-02)	0.136(5.7e-04)
	CG	-	-	-	0.295(3.8e-02)	0.133(8.8e-04)
	Deviation	**3.483**(4.1e-02)	270.760(1.0e+01)	**0.835**(2.6e-03)	**0.228**(1.2e-02)	**0.119**(3.9e-04)

deviation than that of the LOO score. It means that the data points with high model deviations are more effective in reducing the loss of high-loss data points than the data points with high LOO scores. The opposite happens when removing data points from low to high data values (second row). This result matches the result in Fig. 14. Specifically, when $\varepsilon \to 0$, the approximated DRGE is equal to the average loss of the large dataset. In that case, since the density of the high-loss data points is low, data removal using model deviation does not outperform the LOO score significantly. In contrast, when the ε increases, the distributions in $\mathcal{Q}$ are allowed to shift further away from $\hat{P}$ and thus can strategically put more density on the high-loss data points. Consequently, our approach outperforms the LOO score significantly.

Table 6. DRGE (standard error) of the **kernel regression model (KR)** and **NN** trained on data subset (20% of the original dataset size) selected by different baselines. The lower the better.

	Method	HOUSING	UBER	DIABETES	MNIST	CIFAR-10
KR	Random	**2.868**(3.0e-02)	**407.955**(1.1e+01)	0.953(1.1e-02)	2.120(3.4e-02)	2.849(5.7e-02)
	LOO	3.153(6.1e-02)	446.132(3.8e+01)	1.742(2.8e-02)	2.659(6.3e-02)	2.665(2.3e-02)
	TracIn	2.922(1.2e-01)	508.732(6.7e+01)	**0.937**(9.2e-03)	2.241(7.8e-02)	2.872(3.4e-02)
	Influence	3.756(1.9e-01)	468.449(1.8e+01)	1.162(9.4e-02)	2.192(9.7e-02)	2.716(4.4e-02)
	DAVINZ	3.300(1.7e-01)	474.912(2.9e+01)	0.993(2.2e-02)	2.762(2.4e-01)	3.506(8.3e-02)
	LAVA	-	-	-	2.026(4.0e-02)	3.330(1.2e-01)
	CG	-	-	-	1.906(4.2e-02)	2.569(2.1e-02)
	Deviation	4.892(7.7e-01)	557.347(9.4e+01)	1.478(8.3e-02)	**1.845**(4.4e-02)	**2.454**(1.7e-02)
NN	Random	4.948(7.3e-02)	394.949(5.2e+01)	0.863(8.2e-03)	0.301(1.5e-02)	0.125(3.6e-04)
	LOO	4.165(7.3e-02)	290.461(9.1e+00)	0.844(6.8e-03)	0.209(8.6e-03)	0.124(2.5e-04)
	TracIn	5.043(4.8e-02)	360.501(4.8e+01)	0.856(7.8e-03)	0.291(1.7e-02)	0.125(5.1e-04)
	Influence	5.036(1.3e-01)	300.453(1.1e+01)	0.788(3.2e-02)	0.306(2.3e-02)	0.132(1.3e-03)
	DAVINZ	4.428(1.6e-01)	662.362(2.4e+01)	0.962(6.0e-03)	0.359(2.9e-02)	0.171(7.6e-03)
	LAVA	-	-	-	0.280(1.3e-02)	0.216(6.8e-03)
	CG	-	-	-	**0.204**(1.7e-02)	**0.119**(8.5e-05)
	Deviation	**3.838**(7.1e-02)	**289.361**(1.7e+01)	**0.736**(1.5e-02)	0.221(1.2e-02)	0.126(5.3e-04)

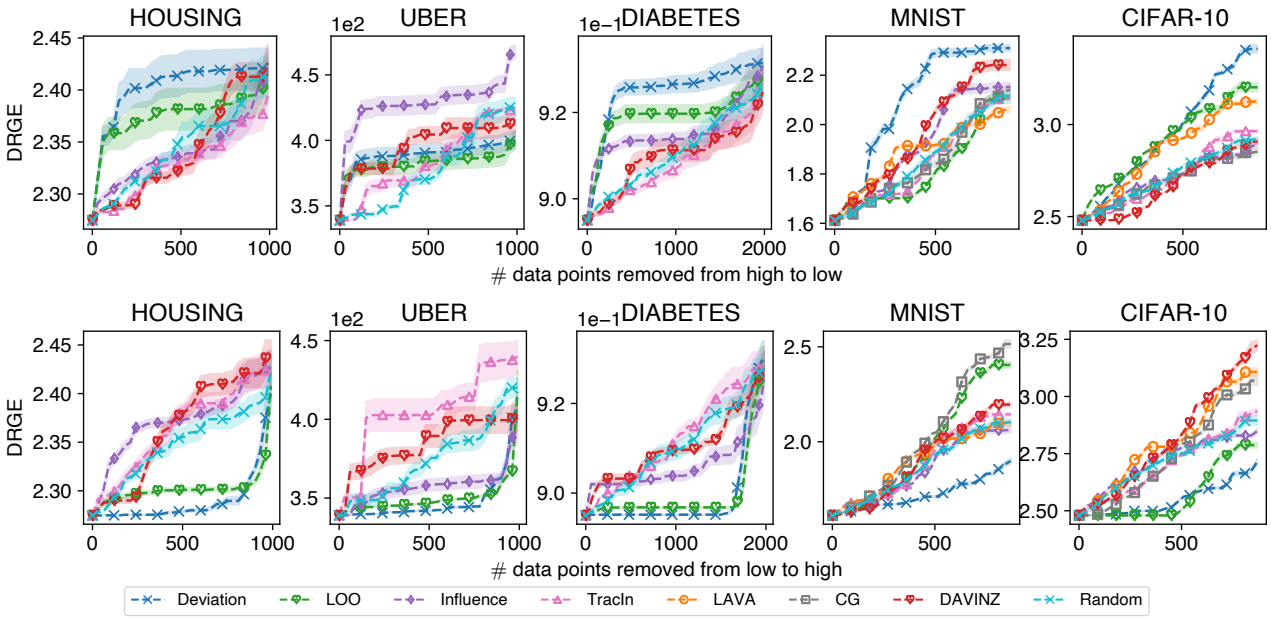

Figure 12. DRGE for **kernel regression model** as data points are removed from the dataset. The first row is removing data points by the data values from high to low, the second row is removing data points by the data values from low to high. We provide the results for all datasets here, as opposed to the selected results shown in Fig. 3.

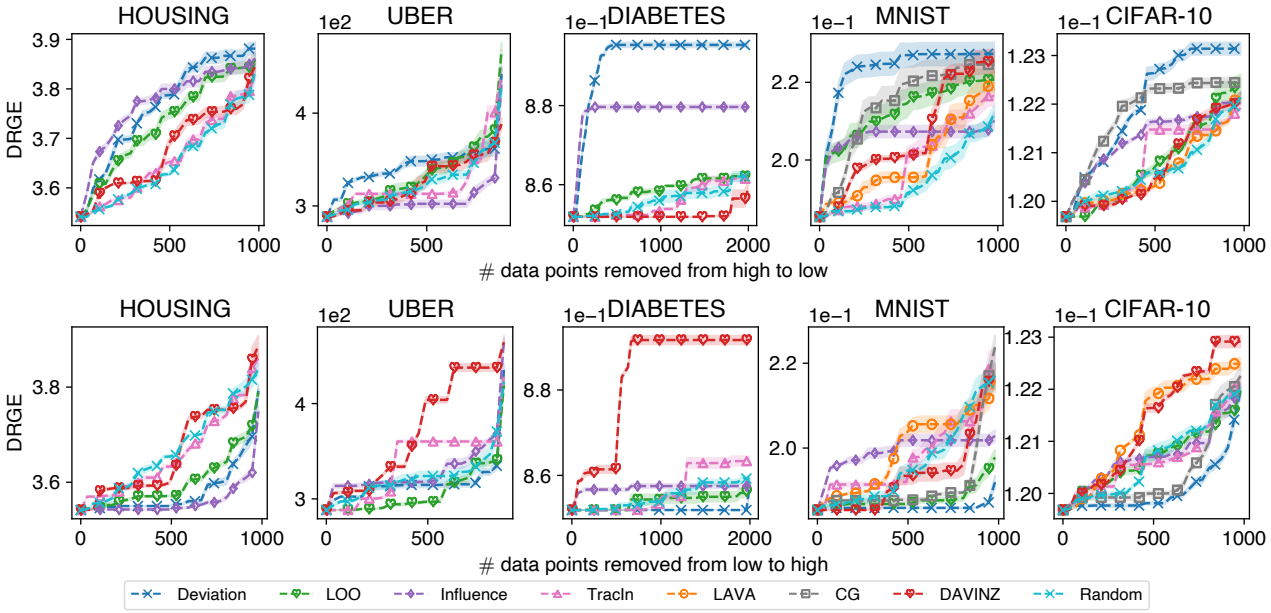

Figure 13. DRGE for **NN** as data points are removed from the dataset. The first row is removing data points by the data values from high to low, the second row is removing data points by the data values from low to high. We provide the results for all datasets here, as opposed to the selected results shown in Fig. 3.

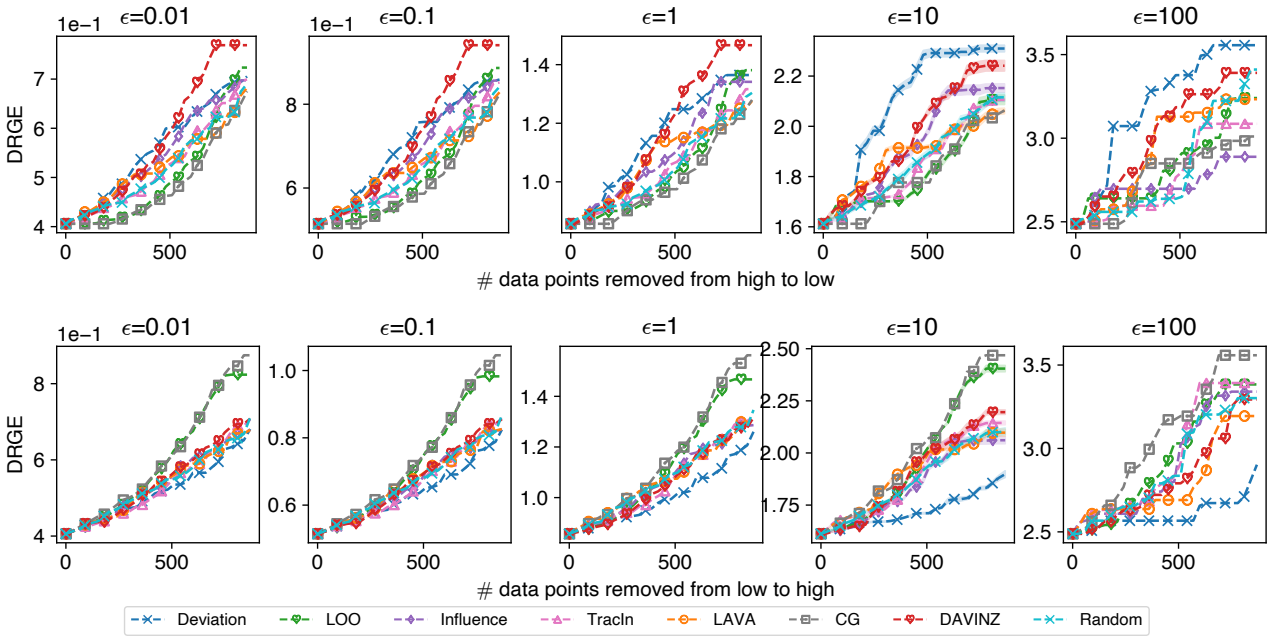

Figure 14. DRGE with different ε for kernel regression model as data points are removed from MNIST. The first row is removing data points by the data values from high to low, the second row is removing data points by the data values from low to high. The results here have more choices of ε compared to Fig. 4.

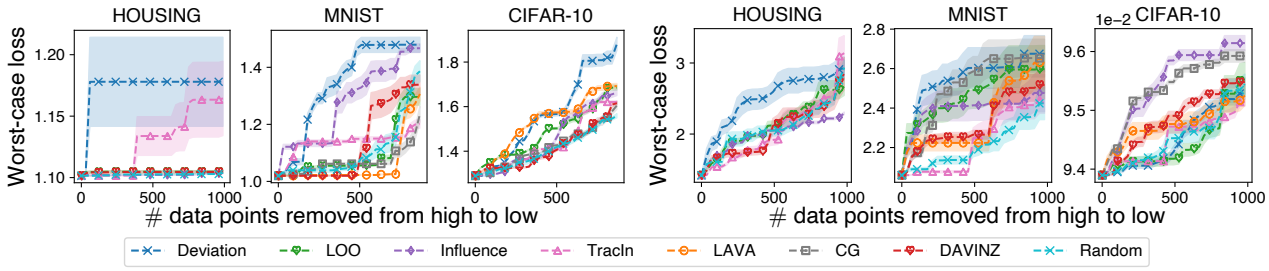

Figure 15. Worst-case performance for kernel regression (first 3 plots) and NN (last 3 plots) as data points are removed by their data values from high to low. We additionally add results for NN here compared to Fig. 5.

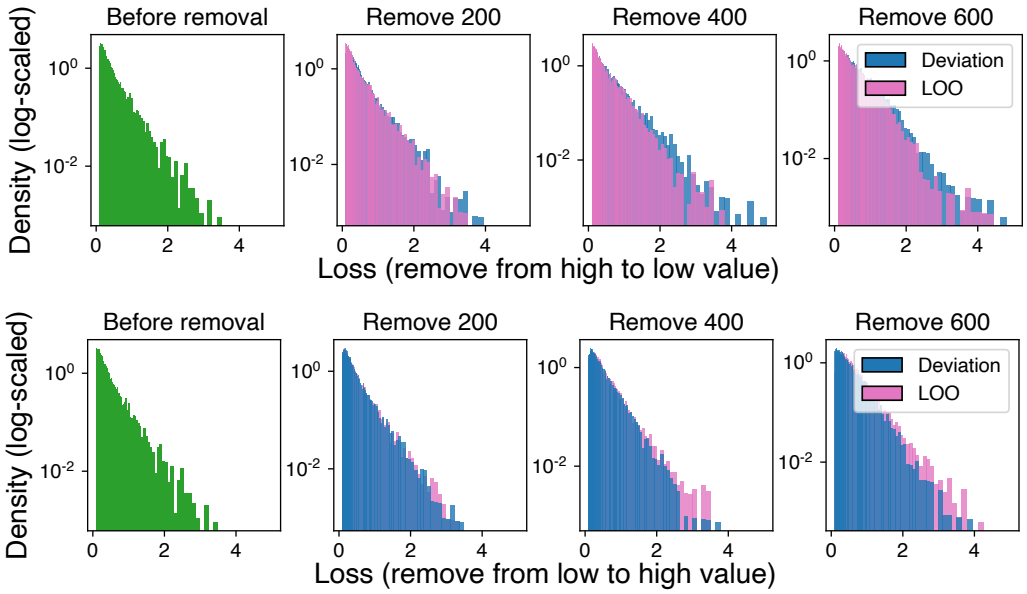

Figure 16. Histogram of pointwise loss (of the large dataset) for the models trained on the datasets with different numbers of data points removed using model deviation and LOO score. The first row is removing data points by the data values from high to low, the second row is removing data points by the data values from low to high.

C. Theoretical Results

C.1. Proof for Theorem 3.3

Proof. We assume that the sampling distribution possesses a non-zero density at all data points within its support, i.e., $p(x) > 0, \forall x \in \mathcal{X}$. For any other distribution Q that has the same support as P and for any function f we have

$$\mathbb{E}_{x\sim Q}\big(f(x)\big) = \mathbb{E}_{x\sim P}\big(\frac{q(x)}{p(x)} f(x)\big) \,.$$

By applying Hoeffding's inequality, we have that with probability at least $1-\delta$,

$$\mathbb{E}_{x\sim Q}\big(f(x)\big) \leq IS_n + b\sqrt{\frac{\ln(1/\delta)}{2n}} \tag{7}$$

where $b = \max_{x\in\mathcal{X}} \frac{q(x)}{p(x)} f(x)$ and $IS_n = \frac{1}{n}\sum_{i=1}^n \frac{q(x_i)}{p(x_i)} f(x_i)$ is the importance sampling estimator of $\mathbb{E}_{x\sim Q}\big(f(x)\big)$ where $x_1, \ldots, x_n$ are i.i.d. samples from the sampling distribution P. The reason why $b = \max_{x\in\mathcal{X}} \frac{q(x)}{p(x)} f(x)$ is that $\frac{q(x)}{p(x)} f(x) \geq 0$ due to the fact that $p(x)$ and $q(x)$ are probability distribution functions and $f(x)$ is the loss function. Therefore, we have that with probability at least $1 - \delta/2$:

$$\begin{aligned}
&R(f_S, \mathcal{Q}) - R(f_{S\cup\{i\}}, \mathcal{Q}) && (8)\\
&= \sup_{Q\in\mathcal{Q}} \mathbb{E}_{x\sim Q}\Big(\ell\big(f_S(x), f^*(x)\big)\Big) - \sup_{Q\in\mathcal{Q}} \mathbb{E}_{x\sim Q}\Big(\ell\big(f_{S\cup\{i\}}(x), f^*(x)\big)\Big) && (9)\\
&\leq \sup_{Q\in\mathcal{Q}} \mathbb{E}_{x\sim Q}\Big(\ell\big(f_S(x), f^*(x)\big) - \ell\big(f_{S\cup\{i\}}(x), f^*(x)\big)\Big) && (10)\\
&\leq \sup_{Q\in\mathcal{Q}} \frac{1}{n}\sum_{j=1}^n \frac{q(x_j)}{p(x_j)}\Big(\ell\big(f_S(x_j), f^*(x_j)\big) - \ell\big(f_{S\cup\{i\}}(x_j), f^*(x_j)\big)\Big) + 2M_1M_2\sqrt{\frac{\ln(2/\delta)}{2n}} && (11)\\
&\leq \sup_{Q\in\mathcal{Q}} \frac{1}{n}\sum_{j=1}^n \frac{q(x_j)}{p(x_j)} L\big(f_S(x_j) - f_{S\cup\{i\}}(x_j)\big) + 2M_1M_2\sqrt{\frac{\ln(2/\delta)}{2n}} && (12)\\
&\leq \sup_{Q\in\mathcal{Q}} \frac{1}{n}\sum_{j=1}^n \frac{q(x_j)}{p(x_j)} L\|f_S - f_{S\cup\{i\}}\|_{\mathcal{H}}\|\Phi(x_j)\|_{\mathcal{H}} + 2M_1M_2\sqrt{\frac{\ln(2/\delta)}{2n}} && (13)\\
&\leq L\|f_S - f_{S\cup\{i\}}\|_{\mathcal{H}} M_0 \sup_{Q\in\mathcal{Q}} \frac{1}{n}\sum_{i=1}^n \frac{q(x_i)}{p(x_i)} + 2M_1M_2\sqrt{\frac{\ln(2/\delta)}{2n}} \,. && (14)
\end{aligned}$$

Inequality (10) is obtained by the additive of suprema. Inequality (11) is obtained by applying probability $1-\delta/2$ and $b = 2M_1M_2$ (since we have $\frac{q(x)}{p(x)} \leq M_1$ and $\ell(f(x), f^*(x)) \leq M_2$) to Inequality (7). Inequality (12) is obtained from the assumption that the loss function ℓ is L-Lipschitz continuous with respect to its first argument. Inequality (13) is obtained from the Cauchy–Schwarz inequality. Inequality (14) is obtained from the assumption that $\|\Phi(x)\|_{\mathcal{H}} \leq M_0, \forall x \in \mathcal{X}$. We

have that with probability at least $1-\delta/2$:

$$\sup_{Q\in\mathcal{Q}} \frac{1}{n}\sum_{i=1}^{n}\frac{q(x_i)}{p(x_i)} = 1 + \sup_{Q\in\mathcal{Q}} \frac{1}{n}\sum_{i=1}^{n}(\frac{q(x_i)}{p(x_i)} - 1) \tag{15}$$

$$\leq 1 + \sup_{Q\in\mathcal{Q}} \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\frac{q(x_i)}{p(x_i)} - 1)^2} \tag{16}$$

$$\leq 1 + \sup_{Q\in\mathcal{Q}} \sqrt{\mathbb{E}_{x\sim P}(\frac{q(x)}{p(x)} - 1)^2 + \max((M_1-1)^2, 1)\sqrt{\frac{\ln(2/\delta)}{2n}}} \tag{17}$$

$$= 1 + \sup_{Q\in\mathcal{Q}} \sqrt{\chi^2(Q,P) + \max((M_1-1)^2, 1)\sqrt{\frac{\ln(2/\delta)}{2n}}} \tag{18}$$

$$= 1 + \sqrt{\sup_{Q\in\mathcal{Q}} \chi^2(Q,P) + \max((M_1-1)^2, 1)\sqrt{\frac{\ln(2/\delta)}{2n}}} \tag{19}$$

$$= 1 + \sqrt{\varepsilon + \max((M_1-1)^2, 1)\sqrt{\frac{\ln(2/\delta)}{2n}}}\,. \tag{20}$$

Inequality (16) is obtained from Cauchy–Schwarz inequality. Inequality (17) is obtained from Hoeffding's inequality by applying the probability $1-\delta/2$ and $b = \max((M_1-1)^2, 1)$ due to the fact that $\frac{q(x)}{p(x)} - 1$ ranges from -1 to $M_1 - 1$. Equ. (18) is obtained from the definition of χ^2 divergence. Equ. (19) is obtained from the monotonically increasing property of the square root function. Equ. (20) is obtained from the definition of $\mathcal{Q}$. By applying union bound on Inequality (14) and Inequality (20), we have that with probability at least $1-\delta$:

$$R(f_S, \mathcal{Q}) - R(f_{S\cup\{i\}}, \mathcal{Q})$$
$$\leq (1 + \sqrt{\varepsilon + \max((M_1-1)^2, 1)\sqrt{\frac{\ln(2/\delta)}{2n}}})LM_0\|f_S - f_{S\cup\{i\}}\|_{\mathcal{H}} + 2M_1M_2\sqrt{\frac{\ln(2/\delta)}{2n}}\,.$$

□

The rationale for using the upper bound derived from Theorem 3.3 as a proxy for the marginal improvement of DRGE. Theorem 3.3 shows that the marginal improvement of DRGE is upper bounded by the model deviation. Therefore we are able to use model deviation as a proxy for the marginal improvement of DRGE. Existing data valuation works (Wu et al., 2022; Nohyun et al., 2022; Just et al., 2023) also commonly adopt the upper bound of the generalization error as (the proxy to) their utility function and demonstrate its effectiveness. For example, DAVINZ(Wu et al., 2022) uses the upper bound of the generalization error of neural networks and LAVA (Just et al., 2023) uses the upper bound for the validation performance. We empirically show that our upper bound can be an effective proxy to the marginal improvement of DRGE from Fig. 1, which shows that a high model deviation indeed induces high marginal improvement of DRGE, and our experimental results in Sec. 5 which show superior performance to other baselines.

C.2. Proof for Theorem 3.4

Our result is based on (Arora et al., 2019, Theorem 3.2). By assuming $|f_0(x_i)| \leq \varepsilon_{\text{init}} \forall \{x_i, y_i\} \in D_S$ and setting $\kappa = 1$, we have the modified version of Theorem 3.2 (Arora et al., 2019). Denote f_S as the NN (i.e., a fully connected NN as defined in Sec. 2) trained on dataset D_S to convergence using gradient descent, and denote f_0 as the NN at initialization. Denote g_S as the minimizer of the kernel regression with NTK Θ trained on D_S. The number of layers in the NN is L_{NN}. The λ_0 (λ_1) is the minimum eigenvalue of the NTK Gram matrix of dataset D_S ($D_{S\cup\{i\}}$). The size of $D_{S\cup\{i\}}$ is n and thus D_S has size $n-1$. We state the modified version of Theorem (Arora et al., 2019) as follows:

Theorem C.1. (Arora et al., 2019, Theorem 3.2) Assume that $\sigma(z) = \max(0, z)$ and $d_1 = d_2 \cdots = d_{L_{\text{NN}}} = m$ and $|f_0(x_i)| \leq \varepsilon_{\text{init}}, \forall \{x_i, y_i\} \in D_S$. Fix a $\varepsilon_k \leq \text{poly}(1/L_{\text{NN}}, 1/(n-1), 1/\log(1/\delta), \lambda_0)$ and set $m \geq \text{poly}(1/\varepsilon_k, n-1, 1/\lambda_0)$. Then for any $x_{\text{test}} \in R^d$ with $\|x_{\text{test}}\| = 1$, with probability at least $1-\delta$ over the random initialization, we have,

$$|f_S(x_{\text{test}}) - g_S(x_{\text{test}})| \leq \varepsilon_k + \varepsilon_{\text{init}}\,. \tag{21}$$

Proof. We apply Theorem C.1 to both f_S and $f_{S\cup\{i\}}$ using union bound. Fix a $\varepsilon_k \leq \min\Big(\text{poly}(1/L_{\text{NN}}, 1/(n-1), 1/\log(1/\delta), \lambda_0), \text{poly}(1/L_{\text{NN}}, 1/n, 1/\log(1/\delta), \lambda_1)\Big)$. Set $m \geq \max\big(\text{poly}(1/\varepsilon_k, n-1, 1/\lambda_0), \text{poly}(1/\varepsilon_k, n, 1/\lambda_1)\big)$. According to Theorem C.1, we have that with probability at least $1-2\delta$ over the random initialization,

$$|f_S(x_{\text{test}}) - g_S(x_{\text{test}})| \leq \varepsilon_k + \varepsilon_{\text{init}}, \quad |f_{S\cup\{i\}}(x_{\text{test}}) - g_{S\cup\{i\}}(x_{\text{test}})| \leq \varepsilon_k + \varepsilon_{\text{init}} \,. \tag{22}$$

For the difference between the DRGE of f_S and the DRGE of g_S, we have the following,

$$\begin{aligned} R(f_S, Q) - R(g_S, Q) &= \sup_{Q\in\mathcal{Q}} \mathbb{E}_{x\sim Q}\ell\big(f_S(x), f^*(x)\big) - \sup_{Q\in\mathcal{Q}} \mathbb{E}_{x\sim Q}\ell\big(g_S(x), f^*(x)\big) \\ &= \mathbb{E}_{x\sim Q_1}\ell\big(f_S(x), f^*(x)\big) - \mathbb{E}_{x\sim Q_2}\ell\big(g_S(x), f^*(x)\big) \\ &\leq \mathbb{E}_{x\sim Q_1}\ell\big(f_S(x), f^*(x)\big) - \mathbb{E}_{x\sim Q_1}\ell\big(g_S(x), f^*(x)\big) \\ &\leq \mathbb{E}_{x\sim Q_1} L|f_S(x) - g_S(x)| \\ &\leq L(\varepsilon_k + \varepsilon_{\text{init}}) \end{aligned} \tag{23}$$

where $Q_1 = \text{argmax}_{Q\in\mathcal{Q}} \mathbb{E}_{x\sim Q}\ell\big(f_S(x), f^*(x)\big)$ and $Q_2 = \text{argmax}_{Q\in\mathcal{Q}} \mathbb{E}_{x\sim Q}\ell\big(g_S(x), f^*(x)\big)$. The first inequality follows from the definition of Q_2. The last inequality follows from the assumption that ℓ is L-Lipschitz continuous with respect to its first argument and Theorem C.1. Similarly, we have:

$$R(f_{S\cup\{i\}}, Q) - R(g_(S\cup\{i\}), Q) \leq L(\varepsilon_k + \varepsilon_{\text{init}}) \,. \tag{24}$$

By applying union bound on Equ. (23), Equ. (24), and Theorem 3.3, we have that with probability at least $1-3\delta$,

$$\begin{aligned} R(f_S, \mathcal{Q}) - R(f_{S\cup\{i\}}, \mathcal{Q}) &= R(g_S, \mathcal{Q}) - R(g_{S\cup\{i\}}, \mathcal{Q}) \\ &\quad + \big(R(f_S, \mathcal{Q}) - R(g_S, \mathcal{Q}) + R(g_{S\cup\{i\}}, \mathcal{Q}) - R(f_{S\cup\{i\}}, \mathcal{Q})\big) \\ &\leq R(g_S, \mathcal{Q}) - R(g_{S\cup\{i\}}, \mathcal{Q}) + 2L(\varepsilon_k + \varepsilon_{\text{init}}) \\ &\leq \kappa_n \|g_S - g_{S\cup\{i\}}\|_{\mathcal{H}} + 2M_1M_2\sqrt{\frac{\ln(2/\delta)}{2n}} + 2L(\varepsilon_k + \varepsilon_{\text{init}}) \end{aligned}$$

where the first inequality follows from Equ. (23) and Equ. (24). The second inequality follows from Theorem 3.3. Replace the 3δ to δ in all equations, we get the final result.

□

C.3. Derivations for Equ. (5) and Equ. (6)

For kernel ridge regression, the closed forms for f_S and $f_{S\cup\{i\}}$ can be written as $f_S = \big((K_S + \lambda I)^{-1}\mathcal{Y}_S\big)^\top K(\mathcal{X}_S, \cdot)$ and $f_{S\cup\{i\}} = \big((K_{S\cup\{i\}} + \lambda I)^{-1}\mathcal{Y}_{S\cup\{i\}}\big)^\top K(\mathcal{X}_{S\cup\{i\}}, \cdot)$. Consequently, we have,

$$\begin{aligned} \mathcal{M}_i(K, D_S)^2 &= \|f_S - f_{S\cup\{i\}}\|^2_{\mathcal{H}_K} \\ &= \langle f_S - f_{S\cup\{i\}}, f_S - f_{S\cup\{i\}}\rangle_{\mathcal{H}_K} \\ &= \langle f_S, f_S\rangle_{\mathcal{H}_K} + \langle f_{S\cup\{i\}}, f_{S\cup\{i\}}\rangle_{\mathcal{H}_K} - 2\langle f_S, f_{S\cup\{i\}}\rangle_{\mathcal{H}_K} \\ &= \boldsymbol{\alpha}^\top K_{S\cup\{i\}}\boldsymbol{\alpha} + \boldsymbol{\beta}^\top K_S\boldsymbol{\beta} - 2\boldsymbol{\beta}^\top K_{S,S\cup\{i\}}\boldsymbol{\alpha} \end{aligned}$$

where $\boldsymbol{\alpha} \coloneqq (K_{S\cup\{i\}} + \lambda I)^{-1}\mathcal{Y}_{S\cup\{i\}}$ and $\boldsymbol{\beta} \coloneqq (K_S + \lambda I)^{-1}\mathcal{Y}_S$. Similarly, the model deviation for NN is $\mathcal{M}_i(\Theta, D_S) = \|g_S - g_{S\cup\{i\}}\|_{\mathcal{H}_\Theta}$ where $g_S = (\Theta_S^{-1}\mathcal{Y}_S)^\top\Theta(\mathcal{X}_S, \cdot)$ and $g_{S\cup\{i\}} = (\Theta^{-1}_{S\cup\{i\}}\mathcal{Y}_{S\cup\{i\}})^\top\Theta(\mathcal{X}_{S\cup\{i\}}, \cdot)$. Therefore,

$$\begin{aligned} \mathcal{M}_i(\Theta, D_S)^2 &= \langle g_S, g_S\rangle_{\mathcal{H}_\Theta} + \langle g_{S\cup\{i\}}, g_{S\cup\{i\}}\rangle_{\mathcal{H}_\Theta} - 2\langle g_S, g_{S\cup\{i\}}\rangle_{\mathcal{H}_\Theta} \\ &= \boldsymbol{\alpha}^\top\Theta_{S\cup\{i\}}\boldsymbol{\alpha} + \boldsymbol{\beta}^\top\Theta_S\boldsymbol{\beta} - 2\boldsymbol{\beta}^\top\Theta_{S,S\cup\{i\}}\boldsymbol{\alpha} \end{aligned}$$

where $\boldsymbol{\alpha} = \Theta^{-1}_{S\cup\{i\}}\mathcal{Y}_{S\cup\{i\}}$ and $\boldsymbol{\beta} = \Theta_S^{-1}\mathcal{Y}_S$.

C.4. Proof for Theorem 3.5

Our result is based on the following result:

Theorem C.2. (Arora et al., 2019, Theorem 3.1) Assume that $\sigma(z) = \max(0, z)$ and the widths of the neural network satisfy $d_l = \Omega(\frac{L_{\text{NN}}^{14}}{\varepsilon_k^4} \log(L_{\text{NN}}/\delta)), \forall l \in \{1, \ldots, L_{\text{NN}}\}$ for a fixed $\varepsilon_k > 0$ and $\delta \in [0, 1]$. Then for $\forall x_1, x_2 \in \mathcal{X}$ that satisfy $\|x_1\| \leq 1, \|x_2\| \leq 1$, we have that with probability at least $1 - \delta$,

$$|\Theta_0(x_1, x_2) - \Theta(x_1, x_2)| \leq (L_{\text{NN}} + 1)\varepsilon_k$$

where $\Theta_0(x_1, x_2) = \nabla_{\theta=\theta_0} f(x_1)^\top \nabla_{\theta=\theta_0} f(x_2)$.

Proof. With Theorem C.2, we are able to give an upper bound of the difference between the model deviation using NTK at initialization Θ_0 and the model deviation using theoretical NTK Θ. For simplicity, denote $[n] := \{1, \ldots, n\}$. Without loss of generality, we assume the data point to be evaluated z_i is the n-th (i.e., the last) data points in the ordered dataset $D_{S\cup\{i\}}$. We denote $\hat{\Theta}_S$ as the NTK at initialization matrix of D_S where $[\hat{\Theta}_S]_{j,k} = \Theta_0(x_j, x_k), \forall x_j, x_k \in D_S$ and denote Θ_S as the NTK matrix of D_S where $[\Theta_S]_{j,k} = \Theta(x_j, x_k), \forall x_j, x_k \in D_S$. Similarly, we denote $\hat{\Theta}_{S\cup\{i\}}$ as the NTK at initialization matrix of $D_{S\cup\{i\}}$ and $\Theta_{S\cup\{i\}}$ as the NTK matrix of $D_{S\cup\{i\}}$. Define $\hat{\Theta}_{S\cup\{i\},S}$ where $[\hat{\Theta}_{S\cup\{i\},S}]_{j,k} = \Theta_0(x_j, x_k), \forall x_j \in D_{S\cup\{i\}}, \forall x_k \in D_S$ and $\Theta_{S\cup\{i\},S}$ where $[\Theta_{S\cup\{i\},S}]_{j,k} = \Theta(x_j, x_k), \forall x_j \in D_{S\cup\{i\}}, \forall x_k \in D_S$.

We can rewrite $\mathcal{M}_i(\Theta, D_S)^2$ by substituting $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ with closed-form expression:

$$\begin{aligned}
&\mathcal{M}_n(\Theta, D_S)^2 \\
&= \mathcal{Y}_S \Theta_S^{-1} \Theta_S \Theta_S^{-1} \mathcal{Y}_S + \mathcal{Y}_{S\cup\{i\}} \Theta_{S\cup\{i\}}^{-1} \Theta_{S\cup\{i\}} \Theta_{S\cup\{i\}}^{-1} \mathcal{Y}_{S\cup\{i\}} - 2\mathcal{Y}_{S\cup\{i\}}^T (\Theta_{S\cup\{i\}}^{-1} \Theta_{S\cup\{i\},S} \Theta_S^{-1}) \mathcal{Y}_S \\
&= \mathcal{Y}_S \Theta_S^{-1} \mathcal{Y}_S + \mathcal{Y}_{S\cup\{i\}} \Theta_{S\cup\{i\}}^{-1} \mathcal{Y}_{S\cup\{i\}} - 2\mathcal{Y}_{S\cup\{i\}}^T (\Theta_{S\cup\{i\}}^{-1} \Theta_{S\cup\{i\},S} \Theta_S^{-1}) \mathcal{Y}_S \,.
\end{aligned}$$

Therefore, we have:

$$\begin{aligned}
&|\mathcal{M}_i(\Theta, D_S)^2 - \mathcal{M}_i(\Theta_0, D_S)^2| \\
\leq &|\mathcal{Y}_S \Theta_S^{-1} \mathcal{Y}_S - \mathcal{Y}_S \hat{\Theta}_S^{-1} \mathcal{Y}_S| + |\mathcal{Y}_{S\cup\{i\}} \Theta_{S\cup\{i\}}^{-1} \mathcal{Y}_{S\cup\{i\}} - \mathcal{Y}_{S\cup\{i\}} \hat{\Theta}_{S\cup\{i\}}^{-1} \mathcal{Y}_{S\cup\{i\}}| + \\
&2|\mathcal{Y}_{S\cup\{i\}}^T (\Theta_{S\cup\{i\}}^{-1} \Theta_{S\cup\{i\},S} \Theta_S^{-1}) \mathcal{Y}_S - \mathcal{Y}_{S\cup\{i\}}^T (\hat{\Theta}_{S\cup\{i\}}^{-1} \hat{\Theta}_{S\cup\{i\},S} \hat{\Theta}_S^{-1}) \mathcal{Y}_S| \\
= &|\mathcal{Y}_S (\Theta_S^{-1} - \hat{\Theta}_S^{-1}) \mathcal{Y}_S| + |\mathcal{Y}_{S\cup\{i\}} (\Theta_{S\cup\{i\}}^{-1} - \hat{\Theta}_{S\cup\{i\}}^{-1}) \mathcal{Y}_{S\cup\{i\}}| + \\
&2|\mathcal{Y}_{S\cup\{i\}}^T (\Theta_{S\cup\{i\}}^{-1} \Theta_{S\cup\{i\},S} \Theta_S^{-1} - \hat{\Theta}_{S\cup\{i\}}^{-1} \hat{\Theta}_{S\cup\{i\},S} \hat{\Theta}_S^{-1}) \mathcal{Y}_S| \,.
\end{aligned} \tag{25}$$

According to Theorem C.2, we have:

$$\|\Theta_{S\cup\{i\}} - \hat{\Theta}_{S\cup\{i\}}\|_2 \leq \|\Theta_{S\cup\{i\}} - \hat{\Theta}_{S\cup\{i\}}\|_F \leq n(L+1)\varepsilon \,.$$

Also, we have:

$$\|\Theta_{S\cup\{i\}}^{-1}\|_2 = \sqrt{\max[\text{eig}(\Theta_{S\cup\{i\}}^{-1})^2]} = \frac{1}{\sqrt{\min[\text{eig}(\Theta_{S\cup\{i\}})^2]}} = \frac{1}{\lambda_{\min}(\Theta_{S\cup\{i\}})} \,.$$

Therefore, we have:

$$\begin{aligned}
\|\Theta_{S\cup\{i\}}^{-1} - \hat{\Theta}_{S\cup\{i\}}^{-1}\|_2 &= \|\Theta_{S\cup\{i\}}^{-1} (\Theta_{S\cup\{i\}} - \hat{\Theta}_{S\cup\{i\}}) \hat{\Theta}_{S\cup\{i\}}^{-1}\|_2 \\
&\leq \|\Theta_{S\cup\{i\}}^{-1}\|_2 \|\Theta_{S\cup\{i\}} - \hat{\Theta}_{S\cup\{i\}}\|_2 \|\hat{\Theta}_{S\cup\{i\}}^{-1}\|_2 \\
&\leq \frac{n(L+1)\varepsilon}{\lambda_{\min}(\Theta_{S\cup\{i\}}) \lambda_{\min}(\hat{\Theta}_{S\cup\{i\}})} \,.
\end{aligned}$$

Similarly, we have $\|\Theta_S^{-1} - \hat{\Theta}_S^{-1}\|_2 \leq \frac{(n-1)(L+1)\varepsilon}{\lambda_{\min}(\Theta_S)\lambda_{\min}(\hat{\Theta}_S)}$.

Assuming that $y \leq B, \forall y \in \mathcal{Y}$, we are able to give an upper bound of the first two terms in the last line of Equ. (25).

$$\begin{aligned}
&|\mathcal{Y}_S(\Theta_S^{-1} - \hat{\Theta}_S^{-1})\mathcal{Y}_S| + |\mathcal{Y}_{S\cup\{i\}}(\Theta_{S\cup\{i\}}^{-1} - \hat{\Theta}_{S\cup\{i\}}^{-1})\mathcal{Y}_{S\cup\{i\}}| \\
&\leq \|\mathcal{Y}_S\|_2\|\Theta_S^{-1} - \hat{\Theta}_S^{-1}\|_2\|\mathcal{Y}_S\|_2 + \|\mathcal{Y}_{S\cup\{i\}}\|_2\|\Theta_{S\cup\{i\}}^{-1} - \hat{\Theta}_{S\cup\{i\}}^{-1}\|_2\|\mathcal{Y}_{S\cup\{i\}}\|_2 \\
&\leq \frac{(n-1)^2B^2(L+1)\varepsilon}{\lambda_{\min}(\Theta_S)\lambda_{\min}(\hat{\Theta}_S)} + \frac{n^2B^2(L+1)\varepsilon}{\lambda_{\min}(\Theta_{S\cup\{i\}})\lambda_{\min}(\hat{\Theta}_{S\cup\{i\}})} \,.
\end{aligned} \tag{26}$$

Denote $[\Theta_{S\cup\{i\},S}]_{n,}$ as the n-th row of the matrix $\Theta_{S\cup\{i\},S}$. Denote $I_S \in \mathbb{R}^{(n-1)\times(n-1)}$ as the identity matrix. Observe that

$$\Theta_{S\cup\{i\},S}\Theta_S^{-1} = \begin{bmatrix} I_S \\ \hline [\Theta_{S\cup\{i\},S}]_{n,}\Theta_S^{-1} \end{bmatrix}; \quad \hat{\Theta}_{S\cup\{i\},S}\hat{\Theta}_S^{-1} = \begin{bmatrix} I_S \\ \hline [\hat{\Theta}_{S\cup\{i\},S}]_{n,}\hat{\Theta}_S^{-1} \end{bmatrix} .$$

From Theorem C.2, we are able to rewrite $[\hat{\Theta}_{S\cup\{i\},S}]_{n,}$ as $[\hat{\Theta}_{S\cup\{i\},S}]_{n,} = [\Theta_{S\cup\{i\},S}]_{n,} + (L+1)\varepsilon\boldsymbol{\nu}_S$, where $\boldsymbol{\nu}_S$ is an $(n-1)$-dimensional row vector with each element satisfies $|[\boldsymbol{\nu}_S]_j| \leq 1, \forall j \in [n-1]$. Therefore, we have:

$$\begin{aligned}
&\|[\Theta_{S\cup\{i\},S}]_{n,}\Theta_S^{-1} - [\hat{\Theta}_{S\cup\{i\},S}]_{n,}\hat{\Theta}_S^{-1}\|_2 \\
&= \big\|[\Theta_{S\cup\{i\},S}]_{n,}\Theta_S^{-1} - \big([\Theta_{S\cup\{i\},S}]_{n,} + (L+1)\varepsilon\boldsymbol{\nu}_S\big)\hat{\Theta}_S^{-1}\big\|_2 \\
&\leq \big\|[\Theta_{S\cup\{i\},S}]_{n,}(\Theta_S^{-1} - \hat{\Theta}_S^{-1})\big\|_2 + (L+1)\varepsilon\|\boldsymbol{\nu}_S\|_2\|\hat{\Theta}_S^{-1}\|_2 \\
&\leq \frac{(n-1)^{3/2}(L+1)\varepsilon}{\lambda_{\min}(\Theta_S)\lambda_{\min}(\hat{\Theta}_S)} + \frac{\sqrt{n-1}(L+1)\varepsilon}{\lambda_{\min}(\hat{\Theta}_S)} \,.
\end{aligned}$$

Denote $C' \coloneqq \frac{(n-1)^{3/2}(L+1)\varepsilon}{\lambda_{\min}(\Theta_S)\lambda_{\min}(\hat{\Theta}_S)} + \frac{\sqrt{n-1}(L+1)\varepsilon}{\lambda_{\min}(\hat{\Theta}_S)}$, then we can rewrite $[\hat{\Theta}_{S\cup\{i\},S}]_{n,}\hat{\Theta}_S^{-1}$ as $[\hat{\Theta}_{S\cup\{i\},S}]_{n,}\hat{\Theta}_S^{-1} = [\Theta_{S\cup\{i\},S}]_{n,}\Theta_S^{-1} + C'\boldsymbol{\nu}'_S$ where $\boldsymbol{\nu}'_S$ is defined similarly to $\boldsymbol{\nu}_S$. Denote $K \coloneqq \Theta_{S\cup\{i\},S}\Theta_S^{-1}$. Denote $\mathbf{0}_S \in \mathbb{R}^{(n-1)\times(n-1)}$ as the matrix with all 0 elements, we have that

$$\hat{\Theta}_{S\cup\{i\},S}\hat{\Theta}_S^{-1} = K + \begin{bmatrix} \mathbf{0}_S \\ \hline C'\boldsymbol{\nu}'_S \end{bmatrix} .$$

Note that:

$$\begin{aligned}
\|K\|_2 = \|\Theta_{S\cup\{i\},S}\Theta_S^{-1}\|_2 = \left\|\begin{bmatrix} I_S \\ \hline [\Theta_{S\cup\{i\},S}]_{n,}\Theta_S^{-1} \end{bmatrix}\right\|_2 &\leq \|I_S\|_2 + \|[\Theta_{S\cup\{i\},S}]_{n,}\Theta_S^{-1}\|_2 \\
&\leq \|I_S\|_2 + \|[\Theta_{S\cup\{i\},S}]_{n,}\|_2\|\Theta_S^{-1}\|_2 \\
&= 1 + \frac{\sqrt{n-1}}{\lambda_{\min}(\Theta_S)} \,.
\end{aligned}$$

Therefore, we can now give an upper bound of the third term in the last line of Equ. (25) as:

$$\begin{aligned}
&2|\mathcal{Y}_{S\cup\{i\}}^T(\Theta_{S\cup\{i\}}^{-1}\Theta_{S\cup\{i\},S}\Theta_S^{-1} - \hat{\Theta}_{S\cup\{i\}}^{-1}\hat{\Theta}_{S\cup\{i\},S}\hat{\Theta}_S^{-1})\mathcal{Y}_S| \\
&\leq 2\|\mathcal{Y}_{S\cup\{i\}}\|_2\|\mathcal{Y}_S\|_2\|\Theta_{S\cup\{i\}}^{-1}\Theta_{S\cup\{i\},S}\Theta_S^{-1} - \hat{\Theta}_{S\cup\{i\}}^{-1}\hat{\Theta}_{S\cup\{i\},S}\hat{\Theta}_S^{-1}\|_2 \\
&= 2\|\mathcal{Y}_{S\cup\{i\}}\|_2\|\mathcal{Y}_S\|_2\left\|\Theta_{S\cup\{i\}}^{-1}K - \hat{\Theta}_{S\cup\{i\}}^{-1}\Big(K + \begin{bmatrix} \mathbf{0}_S \\ \hline C'\boldsymbol{\nu}'_S \end{bmatrix}\Big)\right\|_2 \\
&\leq 2\|\mathcal{Y}_{S\cup\{i\}}\|_2\|\mathcal{Y}_S\|_2\Big(\|K\|_2\|\Theta_{S\cup\{i\}}^{-1} - \hat{\Theta}_{S\cup\{i\}}^{-1}\|_2 + \|\hat{\Theta}_{S\cup\{i\}}^{-1}\|_2\left\|\begin{bmatrix} \mathbf{0}_S \\ \hline C'\boldsymbol{\nu}'_S \end{bmatrix}\right\|_2\Big) \\
&\leq 2\sqrt{n(n-1)}B^2\Big((1 + \tfrac{\sqrt{n-1}}{\lambda_{\min}(\Theta_S)})\tfrac{n(L+1)\varepsilon}{\lambda_{\min}(\Theta_{S\cup\{i\}})\lambda_{\min}(\hat{\Theta}_{S\cup\{i\}})} + \tfrac{C'\sqrt{n-1}}{\lambda_{\min}(\hat{\Theta}_{S\cup\{i\}})}\Big) \\
&\leq 2nB^2\Big(\tfrac{n^{3/2}}{\lambda_{\min}(\Theta_S)\lambda_{\min}(\Theta_{S\cup\{i\}})\lambda_{\min}(\hat{\Theta}_{S\cup\{i\}})} + \tfrac{(n-1)^2}{\lambda_{\min}(\Theta_S)\lambda_{\min}(\hat{\Theta}_S)\lambda_{\min}(\hat{\Theta}_{S\cup\{i\}})}\Big)\big(1 + \lambda_{\min}(\Theta_S)\big)(L+1)\varepsilon \\
&\leq 2n^3B^2\Big(\tfrac{1}{\lambda_{\min}(\Theta_S)\lambda_{\min}(\Theta_{S\cup\{i\}})} + \tfrac{1}{\lambda_{\min}(\Theta_S)\lambda_{\min}(\hat{\Theta}_S)}\Big)\Big(\tfrac{1+\lambda_{\min}(\Theta_S)}{\lambda_{\min}(\hat{\Theta}_{S\cup\{i\}})}\Big)(L+1)\varepsilon \,.
\end{aligned} \tag{27}$$

Therefore, plugging in the results from Equ. (26) and Equ. (27) into Equ. (25), we have,

$$
\begin{aligned}
&|\mathcal{M}_i(\Theta, D_S)^2 - \mathcal{M}_i(\Theta_0, D_S)^2| \\
\leq &|\mathcal{Y}_S(\Theta_S^{-1} - \hat{\Theta}_S^{-1})\mathcal{Y}_S| + |\mathcal{Y}_{S\cup\{i\}}(\Theta_{S\cup\{i\}}^{-1} - \hat{\Theta}_{S\cup\{i\}}^{-1})\mathcal{Y}_{S\cup\{i\}}| + \\
&2|\mathcal{Y}_{S\cup\{i\}}^T(\Theta_{S\cup\{i\}}^{-1}\Theta_{S\cup\{i\},S}\Theta_S^{-1} - \hat{\Theta}_{S\cup\{i\}}^{-1}\hat{\Theta}_{S\cup\{i\},S}\hat{\Theta}_S^{-1})\mathcal{Y}_S| \\
\leq &2n^3B^2\Big(\tfrac{1}{\lambda_{\min}(\Theta_S)\lambda_{\min}(\Theta_{S\cup\{i\}})} + \tfrac{1}{\lambda_{\min}(\Theta_S)\lambda_{\min}(\hat{\Theta}_S)}\Big)\Big(\tfrac{1+\lambda_{\min}(\Theta_S)}{\lambda_{\min}(\hat{\Theta}_{S\cup\{i\}})}\Big)(L+1)\varepsilon \\
&+ \frac{(n-1)^2B^2(L+1)\varepsilon}{\lambda_{\min}(\Theta_S)\lambda_{\min}(\hat{\Theta}_S)} + \frac{n^2B^2(L+1)\varepsilon}{\lambda_{\min}(\Theta_{S\cup\{i\}})\lambda_{\min}(\hat{\Theta}_{S\cup\{i\}})} \\
\leq &4n^3B^2\Big(\tfrac{1}{\lambda_{\min}(\Theta_S)\lambda_{\min}(\Theta_{S\cup\{i\}})} + \tfrac{1}{\lambda_{\min}(\Theta_S)\lambda_{\min}(\hat{\Theta}_S)}\Big)\Big(\tfrac{1+\lambda_{\min}(\Theta_S)}{\lambda_{\min}(\hat{\Theta}_{S\cup\{i\}})}\Big)(L+1)\varepsilon\,.
\end{aligned}
$$

Since we assume that $\min\big(\mathcal{M}_i(\Theta, D_S), \mathcal{M}_i(\Theta_0, D_S)\big) \geq M$, we have

$$
\begin{aligned}
&|\mathcal{M}_i(\Theta, D_S) - \mathcal{M}_i(\Theta_0, D_S)| \\
= &\Big|\frac{\mathcal{M}_i(\Theta, D_S)^2 - \mathcal{M}_i(\Theta_0, D_S)^2}{\mathcal{M}_i(\Theta, D_S) + \mathcal{M}_i(\Theta_0, D_S)}\Big| \\
\leq &\tfrac{2n^3B^2}{M}\Big(\tfrac{1}{\lambda_{\min}(\Theta_S)\lambda_{\min}(\Theta_{S\cup\{i\}})} + \tfrac{1}{\lambda_{\min}(\Theta_S)\lambda_{\min}(\hat{\Theta}_S)}\Big)\Big(\tfrac{1+\lambda_{\min}(\Theta_S)}{\lambda_{\min}(\hat{\Theta}_{S\cup\{i\}})}\Big)(L+1)\varepsilon\,.
\end{aligned}
$$

□

C.5. Proof for Theorem 4.1

Proof. Rewrite $\|\cdot\|_{\mathcal{H}}$ as $\|\cdot\|$ to simplify the proof in the following. Denote the empirical risk of f on the dataset D_S as $\mathrm{er}_S(f) = 1/|D_S| \sum_{(x_m,y_m)\in D_S} \ell(f(x_m), y_m)$. We write the minimization problems for kernel ridge regression on both D_S and $D_{S\cup\{i\}}$ as follows:

$$
\begin{aligned}
f_S &= \operatorname*{argmin}_{g\in\mathcal{H}} \sum_{(x_m,y_m)\in D_S} \ell(g(x_i), y_i) + \frac{\lambda}{2}\|g\|^2\,, \\
f_{S\cup\{i\}} &= \operatorname*{argmin}_{g\in\mathcal{H}} \sum_{(x_m,y_m)\in D_{S\cup\{i\}}} \ell(g(x_i), y_i) + \frac{\lambda}{2}\|g\|^2.
\end{aligned}
$$

By dividing the objective by the number of data points in each dataset, we get the following:

$$
\begin{aligned}
f_S &= \operatorname*{argmin}_{g\in\mathcal{H}} \mathrm{er}_S(g) + \frac{\lambda}{2(n-1)}\|g\|^2\,, \\
f_{S\cup\{i\}} &= \operatorname*{argmin}_{g\in\mathcal{H}} \mathrm{er}_{S\cup\{i\}}(g) + \frac{\lambda}{2n}\|g\|^2.
\end{aligned}
$$

For an $\alpha \in [0, 1]$, we define:

$$
\begin{aligned}
f_\alpha &\coloneqq \alpha f_{S\cup\{i\}} + (1-\alpha) f_S\,, \\
f'_\alpha &\coloneqq (1-\alpha) f_{S\cup\{i\}} + \alpha f_S\,.
\end{aligned}
$$

Since ℓ is convex, for all (x, y), we have:

$$
\begin{aligned}
\ell(f_\alpha(x), y) &\leq \alpha\ell(f_{S\cup\{i\}}(x), y) + (1-\alpha)\ell(f_S(x), y))\,, \\
\mathrm{er}_S(f_\alpha) &\leq \alpha\mathrm{er}_S(f_{S\cup\{i\}}) + (1-\alpha)\mathrm{er}_S(f_S)\,.
\end{aligned} \tag{28}
$$

Since $f_{S\cup\{i\}}$ and f_S are the minimizers of their corresponding minimization problems, we have:

$$
\begin{aligned}
\mathrm{er}_{S\cup\{i\}}(f_{S\cup\{i\}}) + \frac{\lambda}{2n} &\leq \mathrm{er}_{S\cup\{i\}}(f_\alpha) + \frac{\lambda}{2n}\|f_\alpha\|^2\,, \\
\mathrm{er}_S(f_S) + \frac{\lambda}{2(n-1)} &\leq \mathrm{er}_S(f'_\alpha) + \frac{\lambda}{2(n-1)}\|f'_\alpha\|^2\,.
\end{aligned} \tag{29}
$$

Consequently, by rearranging the Inequality (29), we have:

$$\begin{aligned}&\underbrace{\frac{\lambda}{2}(\frac{1}{n}\|f_{S\cup\{i\}}\|^2+\frac{1}{n-1}\|f_S\|^2-\frac{1}{n}\|f_\alpha\|^2-\frac{1}{n-1}\|f'_\alpha\|^2)}_{A}\\&\leq\underbrace{\mathrm{er}_{S\cup\{i\}}(f_\alpha)-\mathrm{er}_{S\cup\{i\}}(f_{S\cup\{i\}})+\mathrm{er}_S(f'_\alpha)-\mathrm{er}_S(f_S)}_{B}\ .\end{aligned}\tag{30}$$

For the LHS, we have:

$$\begin{aligned}A=&\frac{\lambda}{2}(\frac{1}{n}\|f_{S\cup\{i\}}\|^2+(\frac{1}{n}+\frac{1}{n(n-1)})\|f_S\|^2-\frac{1}{n}\|f_\alpha\|^2-(\frac{1}{n}+\frac{1}{n(n-1)})\|f'_\alpha\|^2)\\=&\frac{\lambda}{2n}(\|f_{S\cup\{i\}}\|^2+\|f_S\|^2-\|f_\alpha\|^2-\|f'_\alpha\|^2)+\frac{\lambda}{2n(n-1)}(\|f_S\|^2-\|f'_\alpha\|^2))\\=&\frac{\lambda}{2n}\big(2\alpha(1-\alpha)\|f_{S\cup\{i\}}-f_S\|^2\big)\\&+\frac{\lambda}{2n(n-1)}\big((1-\alpha)\alpha\|f_{S\cup\{i\}}-f_S\|^2+(1-\alpha)(\|f_S\|^2-\|f_{S\cup\{i\}}\|^2)\big)\\=&\frac{\lambda(1-\alpha)}{2n(n-1)}\Big(\big(2n-\alpha\big)\|f_{S\cup\{i\}}-f_S\|^2+(\|f_S\|^2-\|f_{S\cup\{i\}}\|^2)\Big)\\=&\frac{\lambda(1-\alpha)}{2n(n-1)}\Big(\big(2n-\alpha-k\big)\|f_{S\cup\{i\}}-f_S\|^2+k\|f_{S\cup\{i\}}-f_S\|^2+(\|f_S\|^2-\|f_{S\cup\{i\}}\|^2)\Big)\ .\end{aligned}\tag{31}$$

We assume that $\|\Phi(x)\|_{\mathcal{H}}\leq M,\forall x\in\mathcal{X}$ and denote $G:=\max\{\|f_S\|,\|f_{S\cup\{i\}}\|\}$. For the RHS, by applying the Inequality (28), we have:

$$\begin{aligned}B\leq&(1-\alpha)(\mathrm{er}_{S\cup\{i\}}(f_S)-\mathrm{er}_{S\cup\{i\}}(f_{S\cup\{i\}})+\mathrm{er}_S(f_{S\cup\{i\}})-\mathrm{er}_S(f_S))\\=&(1-\alpha)\Big((\mathrm{er}_{S\cup\{i\}}(f_S)--\mathrm{er}_S(f_S))-(\mathrm{er}_{S\cup\{i\}}(f_{S\cup\{i\}})-\mathrm{er}_S(f_{S\cup\{i\}}))\Big)\\=&(1-\alpha)\Big[\Big(\tfrac{1}{n}(\sum_{m=1}^{n-1}\ell(f_S(x_m),y_m)+\ell(f_S(x_i),y_i))-\tfrac{1}{n-1}\sum_{m=1}^{n-1}\ell(f_S(x_m),y_m)\Big)\\&-\Big(\tfrac{1}{n}(\sum_{m=1}^{n-1}\ell(f_{S\cup\{i\}}(x_m),y_m)+\ell(f_{S\cup\{i\}}(x_i),y_i))-\tfrac{1}{n-1}\sum_{m=1}^{n-1}\ell(f_{S\cup\{i\}}(x_m),y_m)\Big)\Big]\\=&(1-\alpha)\Big[\Big(\tfrac{1}{n}\ell(f_S(x_i),y_i)-\tfrac{1}{n(n-1)}\sum_{m=1}^{n-1}\ell(f_S(x_m),y_m)\Big)\\&-\Big(\tfrac{1}{n}\ell(f_{S\cup\{i\}}(x_i),y_i)-\tfrac{1}{n(n-1)}\sum_{m=1}^{n-1}\ell(f_{S\cup\{i\}}(x_m),y_m)\Big)\Big]\\=&\tfrac{(1-\alpha)}{n(n-1)}\sum_{m=1}^{n-1}\Big(\ell(f_S(x_i),y_i)-\ell(f_S(x_m),y_m)-\ell(f_{S\cup\{i\}}(x_i),y_i)+\ell(f_{S\cup\{i\}}(x_m),y_m)\Big)\\=&\tfrac{(1-\alpha)}{n(n-1)}\sum_{(x_m,y_m)\in Z_i}\Big(\ell(f_S(x_i),y_i)-\ell(f_S(x_m),y_i)-\ell(f_{S\cup\{i\}}(x_i),y_i)+\ell(f_{S\cup\{i\}}(x_m),y_i)\Big)\\&+\tfrac{(1-\alpha)}{n(n-1)}\sum_{(x_m,y_m)\notin Z_i}\Big(\ell(f_S(x_i),y_i)-\ell(f_{S\cup\{i\}}(x_i),y_i)+\ell(f_{S\cup\{i\}}(x_m),y_m)-\ell(f_S(x_m),y_m)\Big)\\\leq&\tfrac{(1-\alpha)}{n(n-1)}LG\sum_{(x_m,y_m)\in Z_i}\|\Phi(x_m)-\Phi(x_i)\|+\tfrac{(1-\alpha)}{n(n-1)}2LM(n-m-1)\|f_S-f_{S\cup\{i\}}\|\end{aligned}\tag{32}$$

Denote $I(x_i) \coloneqq \sum_{(x_m,y_m)\in Z_i} \|\Phi(x_m) - \Phi(x_i)\|$. By plugging in results from Equ.(31) and Equ.(32) into Equ.(30), we have:

$$\frac{\lambda}{2}\Big((2n-\alpha-k)\|f_{S\cup\{i\}} - f_S\|^2 + k\|f_{S\cup\{i\}} - f_S\|^2 + (\|f_S\|^2 - \|f_{S\cup\{i\}}\|^2)\Big)$$
$$\leq LGI(x_i) + 2LM(n-m-1)\|f_S - f_{S\cup\{i\}}\| \,.$$

We assume that there exists a k that satisfies $2n-1 \geq k \geq \frac{\|f_{S\cup\{i\}}\|^2 - \|f_S\|^2}{\|f_{S\cup\{i\}} - f_S\|^2}$. Then we have,

$$\frac{\lambda}{2}(2n-\alpha-k)\|f_{S\cup\{i\}} - f_S\|^2 \leq LGI(x_i) + 2LM(n-m-1)\|f_S - f_{S\cup\{i\}}\| \,.$$

Consequently,

$$\frac{\lambda}{2}(2n-\alpha-k)\|f_{S\cup\{i\}} - f_S\|^2 - 2LM(n-m-1)\|f_S - f_{S\cup\{i\}}\| - LGI(x_i) \leq 0 \,.$$

Solving the above quadratic inequality, we obtain,

$$\|f_{S\cup\{i\}} - f_S\| \leq \frac{2LM(n-m-1) + \sqrt{4L^2M^2(n-m-1)^2 + 2\lambda(2n-\alpha-k)LGI(x_i)}}{\lambda(2n-\alpha-k)} \,.$$

□

Remark on the notion of uniqueness that arises from Theorem 4.1. The notion of uniqueness contains dissimilarity and scarcity as we described in Sec. 4. Specifically, data points that are further away from other data points with the same label in the feature space will have higher dissimilarity and thus higher uniqueness which might indicate not very low model deviation. In this sense, our approach does not explicitly consider the setting of anomaly data points since anomaly data points might have high dissimilarity and thus not very low data values (though not necessarily true). This suggests that differentiating between unique, valuable data points and anomalous data points is a potential future direction.

C.6. Extension of Theorem 3.5 and Its Proof

Theorem C.3 (Approximation error of model deviation using NTK at any time step t)**.** For fixed $\varepsilon_k > 0$, assume that for each layer $\forall j \in \{1,\ldots,L_{\text{NN}}\}$, its width $d_j = \Omega(\frac{L_{\text{NN}}^{14}}{\varepsilon_k^4}\log(L_{\text{NN}}/\delta))$ and $d_j > d_{\text{large}}$ such that there exists ε' that $\sup_{t\geq 0}\|\hat{\Theta}_t - \hat{\Theta}_0\|_F \leq \varepsilon'/{d_j}^{1/2}$. Assume that $\lambda_{\min}(\Theta_{S\cup\{i\}}) > 0$, each label $\forall y \in \mathcal{Y}, y \leq B$, and each input $\forall x \in \mathcal{X}, \|x\| \leq 1$. Apply gradient descent with learning rate $\eta \leq 2(\lambda_{\min}(\Theta_{S\cup\{i\}}) + \lambda_{\max}(\Theta_{S\cup\{i\}}))^{-1}$. Then, $\forall\delta \in (0,1)$, with probability at least $1-\delta$,

$$|\mathcal{M}_i(\Theta, D_S) - \mathcal{M}_i(\Theta_t, D_S)| \leq C\big((L_{\text{NN}}+1)\varepsilon_k + \varepsilon'/d_j^{1/2}\big)$$

where $C = \frac{2n^3B^2}{M}\big(\frac{1}{\lambda_{\min}(\Theta_S)\lambda_{\min}(\Theta_{S\cup\{i\}})} + \frac{1}{\lambda_{\min}(\Theta_S)\lambda_{\min}(\hat{\Theta}_S)}\big)\big(\frac{1+\lambda_{\min}(\Theta_S)}{\lambda_{\min}(\hat{\Theta}_{S\cup\{i\}})}\big)$ and $M = \min(\mathcal{M}_i(\Theta, D_S), \mathcal{M}_i(\Theta_t, D_S))$.

Theorem C.3 provides the approximation error of model deviation using NTK at any time step t instead of NTK at initialization, bringing another flexibility to our approach. The proof for Theorem C.3 is based on the following result:

Theorem C.4. (Lee et al., 2019, Theorem 2.1) Let $d_1 = \cdots = d_{L_{\text{NN}}} = d$ and assume $\lambda_{\min}(\Theta) > 0$. Applying gradient decent with learning rate $\eta < 2(\lambda_{\min}(\Theta_{S\cup\{i\}}) + \lambda_{\max}(\Theta_{S\cup\{i\}}))^{-1}$, for every $x \in \mathcal{X}$ with $\|x\|_2 \leq 1$, with probability arbitrarily close to 1 over random initialization,

$$\sup_{t\geq 0}\|\hat{\Theta}_t - \hat{\Theta}_0\|_F = \mathcal{O}(d^{-1/2}), \quad \text{as } d \to \infty \,.$$

Proof. According to Theorem C.4, there exists a d_{large} and ε' such that $\sup_{t\geq 0}\|\hat{\Theta}_t - \hat{\Theta}_0\|_F \leq \varepsilon'/{d_j}^{1/2}$ holds $d_j \geq d_{\text{large}}$. Applying this result to the proof for Theorem 3.5, we obtain the result in Theorem C.3. □

D. Additional Discussion on Related Work

In Sec. 6, we discuss some related works on data valuation. There are some related works on defining the uncertainty set (Namkoong & Duchi, 2017; Gotoh et al., 2018; Staib & Jegelka, 2019; Blanchet et al., 2022). Specifically, (Namkoong & Duchi, 2017; Gotoh et al., 2018) define the uncertainty set using χ^2-divergence on the empirical distribution. However, as we discussed in Appendix A, it can not handle the out-of-sample problem. (Blanchet et al., 2022) defines the uncertainty set using Wasserstein distance which might not be able to model the real-world complex dataset due to the use of easy ground metrics (Staib & Jegelka, 2019) (e.g., Euclidean distance). More discussions can be found in Appendix A. (Staib & Jegelka, 2019) propose to use the maximum mean discrepancy (MMD) as the distance measure to define the uncertainty set to tackle the problems of χ^2-divergence and Wasserstein distance. However, it is dependent on the choice of kernel and thus a poor choice of kernel might result in a poor measure of distributional distance. Additionally, the approximation of worst-case performance (provided in (Staib & Jegelka, 2019)) based on the MMD uncertainty set is only applicable to kernel-based algorithms. Unlike the approaches discussed above, our definition of the uncertainty set applies the χ^2-divergence based on the sampling distribution P, rather than the empirical distribution $\hat{P}$. This approach effectively tackles the out-of-sample problem. Furthermore, our definition of the uncertainty set is not dependent on kernel selection, thereby eliminating the risks associated with poor kernel choices. Our approach utilizes model deviation as a proxy for the marginal improvement of DRGE and can be applied to both kernel-based algorithms and neural networks, not just to kernel-based algorithms.