Supplementary Materials for Towards Efficient, Fair, and Interpretable Neural Dynamic Data Valuation

Anonymous Author

[Affiliation hidden for blind review].

Abstract

Enclosed are the comprehensive supplementary materials for our paper, titled 'Towards Efficient, Fair, and Interpretable Neural Dynamic Data Valuation'.

Contents

1 Nomenclature **2**

2 Related works **2**

2.1 Dynamics and optimal control theory 2

2.2 Data valuation 2

3 Problem formulation **4**

3.1 Loo metric 4

3.2 Static marginal contribution 4

3.3 Shapley value metric 5

4 The axioms of existing methods **5**

5 Details of learning stochastic dynamic **5**

5.1 Basic MSA 5

6 Derivation of re-weighting strategy **5**

6.1 Convergence proof of the training loss 5

6.2 Convergence proof of the meta loss 8

6.3 Derivation of the update details for meta parameters 11

7 Detailed description of the weighted mean-field MSA **12**

7.1 Weighted mean-field MSA 12

7.2 Error estimate for the weighted mean-field MSA 12

8 Proof of NDDV **18**

8.1 Proof of the data state utility function . . . 18
8.2 Proof of dynamic marginal contribution . . . 21
8.3 Proof of dynamic data valuation . . . 21

9 Experiments details 24
9.1 Experimental Setup . . . 24
9.2 Datasets . . . 25
9.3 Hyperparameters for existing methods . . . 25
9.4 Pseudo algorithm . . . 27
9.5 Convergence verification for the training loss . . . 28
9.6 Additional results on efficiency and effectiveness . . . 28
9.7 Additional results on the data valuation process . . . 29
9.8 Additional results on interpretability . . . 29
9.9 Ablation study . . . 29

1 Nomenclature

Table S1 presents the key abbreviations and notations employed throughout this work.

2 Related works

2.1 Dynamics and optimal control theory

The dynamical perspective, viewing Deep Neural Networks (DNNs) as continuous-time dynamical systems, has provided new insights into their architecture and training processes [1]. This method equates the propagating rule in deep residual networks with a one-step discretization of the forward Euler scheme on an ordinary differential equation (ODE), enhancing numerical approximations in residual blocks [2, 3]. Furthermore, the deep continuum limit allows for transport analysis using Wasserstein geometry [4]. This analogy extends to optimization and control mechanisms in algorithms, promoting new mean-field optimal control problem frameworks for reinterpreting supervised learning methodologies [5, 6]. Inspired by control theory, new training algorithms have been developed [7–9]. A similar analysis can be applied to stochastic gradient descent (SGD) by viewing it as a stochastic dynamical system. Most previous discussions on implicit bias formulate SGD as stochastic Langevin dynamics [10]. Recent studies also consider other stochastic models like Lèvy processes [11]. Stability analyses of Gram matrix dynamics induced by DNNs indicate global optimality [12, 13], leading to optimal adaptive strategies for tuning hyper-parameters in SGD, such as learning rate, momentum, and batch size [14, 15].

2.2 Data valuation

Existing data valuation methods, such as LOO [16], DataShapley [17, 18], BetaShapley [19], DataBanzhaf [20], InfluenceFunction [21] and so on, generally require knowledge of the underlying learning algorithms and are known for their substantial computational demands. Notably, the work by [22] proposes a k-Nearest Neighbor classifier as a learning-agnostic proxy model for data valuation. Still, it is less effective and efficient than NDDV in distinguishing data quality. Alternatively, measuring the utility of a dataset by the volume[23] provides an algorithm-agnostic calculation but fails to detect label errors. Assessing data points with the Shapley value is still costly for large datasets. Moreover, the work by [24] introduces a learning-agnostic data valuation method using class-wise Wasserstein distances, yet it relies on a validation set and has limitations in assessing fairness. Recently, a method based on the out-of-bag estimate is computationally efficient and outperforms existing methods [25]. Despite its advantages, this method is constrained by the sequential dependency of weak learners in boosting, limiting its direct applicability to downstream tasks. Marginal contribution-based methods have been studied and applied to various machine learning problems, including feature attribution [26, 27], model explanation [28, 29], and collaborative learning [30, 31]. Among these works, the Shapley value is one of the most widely

Table S1: Abbreviations and notations.

LOO	the leave-one-out
OOB	the out-of-bag
MLP	a multi-layer perceptron network
FSDEs	the Forward Stochastic Differential Equations
BSDEs	the Backward Stochastic Differential Equations
FBSDEs	the Forward-Backward Stochastic Differential Equations
SMP	the Stochastic Maximum Principle
MSA	the method of successive approximations
TPR	the true positive rate
FPR	the false positive rate
EOp-score	the equal opportunity score
EOdds-score	the equalized odds score
KANs	the Kolmogorov–Arnold Networks
RBF	the radial basis function
x_i	a data point
y_i	a label corresponding x_i
$[N]$	a training set of size N
$\mathcal{D}$	the training dataset
S	a subset of the training dataset $\mathcal{D}$
$\mathcal{X}$	the input space belongs to $\mathbb{R}^d$
$\mathcal{Y}$	the label space belongs to $\mathbb{R}$
U	a utility function belongs to $2^N \to \mathbb{R}$
$U_i(S)$	the data state utility for the data point (x_i, y_i) in subset S
$\mathcal{A}$	a base model trained on the subset S
$\Delta_j(x_i, y_i;\, U)$	the static marginal contribution of (x_i, y_i) for a utility function U and $j \in [N]$
$\Delta(x_i, y_i;\, U_i(S))$	the dynamic marginal contribution of (x_i, y_i)
$\phi_{\text{loo}}(x_i, y_i;\, U)$	the value of data point (x_i, y_i) is evaluated using the LOO metric
$\phi_{\text{Shap}}(x_i, y_i;\, U)$	the value of data point (x_i, y_i) is evaluated using the Shapley value metric
$\phi(x_i, y_i;\, U_i(S)$	the value of data point (x_i, y_i) is evaluated using NDDV
ψ	the stochastic control parameters
X_t	the data state
Y_t	the data co-state
Z_t	the coefficient of W_t
Φ	the terminal cost function, corresponding to the typical loss term
R	the running cost function, playing a role in regularization
b	the drift function, embodying a combination of a linear transformation
σ	the diffusion function, remaining identical constants for all t
W_t	the standard Wiener process and $\mathrm{d}W_t$ remains identical constants for all t
H	the Hamiltonian
$\mathcal{V}$	the meta-weight function
θ	the hyper-parameters in the meta-network
ℓ	the meta loss
K_ν	the modified Bessel function
Γ	the Gamma function
h_b	a basis function, utilizing the SiLU activation function
h_k	parametrized function as RBF
α_b	the weighted parameter via the Xavier initialization for h_b
α_k	the trainable coefficient for h_k

used marginal contribution-based methods, and many alternative methods have been studied by relaxing some of the underlying fair division axioms [32, 33]. Additionally, some methods are independent of marginal contributions. For instance, in the data valuation literature, a data value estimator model using reinforcement learning is proposed by [34]. This method combines data valuation with model training using reinforcement learning. However, it measures data usage likelihood, not the impact on model quality.

To our knowledge, optimal control has not yet been exploited for data valuation problems. The proposed NDDV method diverges from existing methods in several key aspects. First, we construct a data valuation framework from the perspective of optimal control, marking a pioneering endeavor in this field. Second, whereas most data valuation methods rely on calculating marginal contributions within cooperative games, necessitating repetitive training of a predefined utility function, our method derives control strategies from a continuous time optimization process. The gradient of the Hamiltonian concerning the control states serves as the marginal contribution, representing a novel attempt at data valuation problems. Finally, we transform the interactions among data points into interactions between data points and the mean-field state, thereby circumventing the need for exponential combinatorial computations.

3 Problem formulation

In this section, we formally describe the data valuation problem. Then, we review the concept of marginal contribution-based data valuation.

In various downstream tasks, data valuation aims to fairly assign model performance scores to each data point, reflecting the contribution of individual data points. Let $[N] = \{1, \ldots, N\}$ denotes a training set of size n. We define the training dataset as $\mathcal{D} = (x_i, y_i)_{i=1}^{N}$, where each pair (x_i, y_i) consists of an input x_i from the input space $\mathcal{X} \subset \mathbb{R}^d$ and a corresponding label y_i from the label space $\mathcal{Y} \subset \mathbb{R}$, pertaining to the i-th data point. To measure the contributions of data points, we define a utility function $U : 2^N \to \mathbb{R}$, which takes a subset of the training dataset $\mathcal{D}$ as input and outputs the performance score that is trained on that subset. In classification tasks, for instance, a common choice for U is the test classification accuracy of an empirical risk minimizer trained on a subset of $\mathcal{D}$. Formally, we set $U(S) = \texttt{metric}(\mathcal{A}(S))$, where $\mathcal{A}$ denotes a base model trained on the dataset S, and `metric` represents the metric function for evaluating the performance of $\mathcal{A}$, e.g., the accuracy of a finite hold-out validation set. When $S = \{\}$ is the empty set, $U(S)$ is set to be the performance of the best constant model by convention. The utility function is influenced by the selection of learning algorithms and a specific class. However, this dependency is omitted in our discussion, prioritizing the comparative analysis of data value formulations instead. For a set S, we denote its power set by 2^S and its cardinality by $|S|$. We set $[j] := \{1, \ldots, j\}$ for $j \in \mathbb{N}$.

A standard method for quantifying data values is the marginal contribution, which measures the average change in a utility function when a particular datum is removed from a subset of the entire dataset $\mathcal{D}$. We denote the data value of data point $(x_i, y_i) \in \mathcal{D}$ computed from U as $\phi(x_i, y_i;\ U)$. The following sections review well-known notions of data value.

3.1 Loo metric

A simple data value measure is the LOO metric, which calculates the change in model performance when the data point (x_i, y_i) is excluded from the training set N

$$\phi_{\mathrm{loo}}(x_i, y_i;\ U) \triangleq U(N) - U(N \setminus (x_i, y_i)). \tag{1}$$

The LOO metric quantifies the impact of removing a specific data point (x_i, y_i) from the entire dataset $\mathcal{D}$. This metric incorporates Cook's distance and the approximate empirical influence function to measure changes. However, it is known to be computationally feasible, but it often assigns erroneous values that are close to zero [35].

3.2 Static marginal contribution

Existing standard data valuation methods can be expressed as a function of the marginal contribution. For a specific utility function U and $j \in [N]$, the marginal contribution of $(x_i, y_i) \in \mathcal{D}$ with respect to $[j]$ data points is defined as follows

$$\Delta_j(x_i, y_i;\ U) \triangleq \frac{1}{\binom{N-1}{j-1}} \sum_{S \in \mathcal{D}^{\setminus(x_i, y_i)}} U(S \cup (x_i, y_i)) - U(S), \tag{2}$$

where $\mathcal{D}^{\setminus(x_i,y_i)} = \{S \subseteq \mathcal{D}\setminus(x_i, y_i) : |S| = j - 1\}$. Eq.(2) is a combination to calculate the error in adding (x_i, y_i), which is a static metric.

3.3 Shapley value metric

The Shapley value metric is considered the most widely studied data valuation scheme, originating from cooperative game theory. At a high level, it appraises each point based on the average utility change caused by adding the point into different subsets. The Shapley value of a data point i is defined as

$$\phi_{\text{Shap}}(x_i, y_i;\, U) \triangleq \frac{1}{N} \sum_{j=1}^{N} \Delta_j(x_i, y_i;\, U). \tag{3}$$

As its extension, Beta Shapley is proposed by is expressed as a weighted mean of marginal contributions

$$\phi_{\text{Beta}}(x_i, y_i;\, U) \triangleq \sum_{j=1}^{N} \beta_j \Delta_j(x_i, y_i;\, U). \tag{4}$$

where $\beta = (\beta_1, \ldots, \beta_n)$ is a predefined weight vector such that $\sum_{j=1}^{N} \beta_j = 1$ and $\beta_j \geq 0$ for all $j \in [N]$. A functional form of Eq.(4) is also known as semi-values.

Shapley value metric is empirically more effective than the LOO metric in many downstream tasks such as mislabeled data detection in classification settings [18, 19]. However, their computational complexity is well known to be expensive, making it infeasible to apply to large datasets [17, 20, 36]. As a result, most existing data valuation methods have focused on small datasets, *e.g.*, $n \leq 1,000$.

4 The axioms of existing methods

The popularity of the Shapley value is attributable to the fact that it is the unique data value notion satisfying the following five common axioms [27], as follow

- **Efficiency**: The values add up to the difference in value between the grand coalition and the empty coalition: $\sum_{i \in N} \phi(x_i, y_i;\, U) = U(N) - U(\emptyset)$;
- **Symmetry**: if $U(S \cup (x_i, y_i)) = U(S \cup (x_j, y_j))$ for all $S \in N \setminus \{(x_i, y_i), (x_j, y_j)\}$, then $\phi(x_i, y_i;\, U) = \phi(x_j, y_j;\, U)$;
- **Dummy**: if $U\left(S \cup (x_i, y_i)\right) = U(S)$ for all $S \in N \setminus (x_i, y_i)$, then it makes 0 marginal contribution, so the value of the data point (x_i, y_i) should be $\phi\left(x_i, y_i;\, U\right) = 0$;
- **Additivity**: For two utility functions U_1, U_2 and arbitrary $\alpha_1, \alpha_2 \in \mathbb{R}$, the total contribution of a data point (x_i, y_i) is equal to the sum of its contributions when combined: $\phi\left((x_i, y_i);\, \alpha_1 U_1(S) + \alpha_2 U_2(S)\right) = \alpha_1 \phi\left(x_i, y_i;\, U_1(S)\right) + \alpha_2 \phi\left(x_i, y_i;\, U_2(S)\right)$;
- **Marginalism**: For two utility functions U_1, U_2, if each data point has the identical marginal impact, they receive same valuation: $U_1(S \cup (x_i, y_i)) - U_1(S) = U_2(S \cup (x_i, y_i)) - U_2(S)$ holds for all $((x_i, y_i);\, S)$, then it holds that $\phi(x_i, y_i;\, U_1) = \phi(x_i, y_i;\, U_2)$.

5 Details of learning stochastic dynamic

5.1 Basic MSA

In this section, we illustrate the iterative update process of the basic MSA in Alg.S1, comprising the state equation, the costate equation, and the maximization of the Hamiltonian.

6 Derivation of re-weighting strategy

6.1 Convergence proof of the training loss

Lemma 1 Consider two non-negative real sequences $(a_n)_{n \leq 1}$ and $(b_n)_{n \leq 1}$ with the following properties:

- The series $\sum_{n=1}^{\infty} a_n$ diverges.
- The series $\sum_{n=1}^{\infty} a_n b_n$ converges.

Algorithm S1 Basic MSA

```
Initialize $\psi^0 = \{\psi_t^0 \in \Psi : t = 0 \dots, T-1\}$.
for $k = 0$ to $K$ do
    Solve the forward SDE:
        $$\mathrm{d}X_t^k = b(t, X_t^k, \psi_t^k)\mathrm{d}t + \sigma \mathrm{d}W_t, \quad X_0^k = x,$$
    Solve the backward SDE:
        $$\mathrm{d}Y_t^k = -\nabla_x \mathcal{H}\big(t, X_t^k, Y_t^k, \psi_t^k)\mathrm{d}t + Z_t^k \mathrm{d}W_t, \quad Y_T^k = -\nabla_x \Phi(X_T^k, \psi_T^k),$$
    For each $t \in [0, T-1]$, update the state control $\psi_t^{k+1}$:
        $$\psi_t^{k+1} = \arg\max_{\psi \in \Psi} \mathcal{H}(t, X_t^k, Y_t^k, Z_t^k, \psi).$$
end for
```

- There exists a constant $K > 0$ such that $|b_{n+1} - b_n| \leq K a_n$ for all n.

Under these conditions, the sequence $(b_n)_{n \geq 1}$ converges to 0.

Theorem 1 Let $\mathcal{L}$ be a Lipschitz smooth loss function with Lipschitz constant L, and let $\mathcal{V}(\cdot)$ be a differentiable function with a δ-bounded gradient and a twice-differentiable Hessian bounded by $\mathcal{B}$. Assume that $\mathcal{L}$ has ρ-bounded gradients for training and metadata points. Suppose the learning rate α^k is defined as $\alpha^k = \min\left\{1, \frac{k}{T}\right\}$, for some constant $k > 0$ such that $\frac{k}{T} < 1$. Let β^k be a monotone decreasing sequence with $\beta^k = \min\left\{\frac{1}{L}, \frac{c}{\sigma\sqrt{T}}\right\}$ for some constant $c > 0$, ensuring that $\frac{\sigma\sqrt{T}}{c} \geq L$ and $\sum_{k=1}^{\infty} \beta^k < \infty$, $\sum_{k=1}^{\infty} (\beta^k)^2 < \infty$. Then,

$$\lim_{k \to \infty} \mathbb{E}\left[\|\nabla \mathcal{L}(\psi^k;\ \theta^{k+1})\|^2\right] = 0. \tag{5}$$

Proof It is easy to conclude that α^k satisfy $\sum_{k=0}^{\infty} \alpha^k = \infty, \sum_{k=0}^{\infty} (\alpha^k)^2 < \infty$. Recall the update of ψ in each iteration as follows

$$\psi^{k+1} = \psi^k + \frac{\alpha}{N} \sum_{i=1}^{N} \nabla_\psi \mathcal{H}_i(\cdot, \psi, \mathcal{V}(\Phi_i(\cdot, \psi_T^k);\ \theta^{k+1}))|_{\psi^k}. \tag{6}$$

It can be written as

$$\psi^{k+1} = \psi^k + \alpha^k \nabla \mathcal{H}(\cdot, \psi^k;\ \theta^{k+1})|_{\Upsilon^k}, \tag{7}$$

where $\nabla \mathcal{H}(\cdot, \psi^k;\ \theta) = -\nabla \mathcal{L}(\psi^k;\ \theta)$. Since the mini-batch υ^k is drawn uniformly at random, we can rewrite the update equation as:

$$\psi^{k+1} = \psi^k + \alpha^k \left[\nabla \mathcal{H}(\cdot, \psi^k;\ \theta^{k+1}) + \upsilon^k\right], \tag{8}$$

where $\upsilon^k = \nabla \mathcal{H}(\cdot, \psi^k;\ \theta^{k+1})|_{\Upsilon^k} - \nabla \mathcal{H}(\cdot, \psi^k;\ \theta^{k+1})$. Note that υ^k is i.i.d. random variable with finite variance since Υ^k are drawn i.i.d. with a finite number of samples. Furthermore, $\mathbb{E}[\upsilon^k] = 0$, since samples are drawn uniformly at random, and $\mathbb{E}[\|\upsilon^k\|^2] \leq \sigma^2$.

The Hamiltonian $\mathcal{H}(\cdot, \psi;\ \theta)$ can be easily checked to be Lipschitz-smooth with constant L, and have ρ-bounded gradients concerning training data. Observe that

$$\begin{aligned} &\mathcal{H}(\cdot, \psi^{k+1};\ \theta^{k+2}) - \mathcal{H}(\cdot, \psi^k;\ \theta^{k+1}) \\ =&\Big\{\mathcal{H}(\cdot, \psi^{k+1};\ \theta^{k+2}) - \mathcal{H}(\cdot, \psi^{k+1};\ \theta^{k+1})\Big\} + \Big\{\mathcal{H}(\cdot, \psi^{k+1};\ \theta^{k+1}) - \mathcal{H}(\cdot, \psi^k;\ \theta^{k+1})\Big\}. \end{aligned} \tag{9}$$

For the first term,

$$\begin{aligned} &\mathcal{H}(\cdot, \psi^{k+1};\ \theta^{k+2}) - \mathcal{H}(\cdot, \psi^{k+1};\ \theta^{k+1}) \\ =&\frac{1}{N} \sum_{i=1}^{N} \sum_{t=0}^{T-1} \Big\{\Big[\mathcal{V}(\Phi_i(\cdot, \psi_T^k);\ \theta^{k+2}) - \mathcal{V}(\Phi_i(\cdot, \psi_T^k);\ \theta^{k+1})\Big]\Big[b(\psi_t^k) Y_t(\psi_t^k) + \sigma Z_t(\psi_t^k)\Big] - R_i(\psi_t^k)\Big\} \end{aligned}$$

$$
\begin{aligned}
&\leq \frac{1}{N}\sum_{i=1}^{N}\sum_{t=0}^{T-1}\Bigg\{\Bigg[\langle \frac{\partial \mathcal{V}(\Phi_i(\cdot,\psi_T^k);\ \theta)}{\partial\theta}\Big|_{\theta^k}, \theta^{k+1}-\theta^k\rangle \\
&\quad +\frac{\delta}{2}\|\theta^{k+1}-\theta^k\|^2\Bigg]\Big[b(\psi_t^k)Y_t(\psi_t^k)+\sigma Z_t(\psi_t^k)\Big]-R_i(\psi_t^k)\Bigg\} \\
&=\frac{1}{N}\sum_{i=1}^{N}\sum_{t=0}^{T-1}\Bigg\{\Bigg[\langle \frac{\partial \mathcal{V}(\Phi_i(\cdot,\psi_T^k);\ \theta)}{\partial\theta}\Big|_{\theta^k}, -\beta^k\Big[\nabla\ell(\hat{\psi}^k(\theta^k))+\xi^k\Big]\rangle \\
&\quad +\frac{\delta\beta_k^2}{2}\|\nabla\ell(\hat{\psi}^k(\theta^k))+\xi^k\|^2\Bigg]\Big[b(\psi_t^k)Y_t(\psi_t^k)+\sigma Z_t(\psi_t^k)\Big]-R_i(\psi_t^k)\Bigg\} \\
&=\frac{1}{N}\sum_{i=1}^{N}\sum_{t=0}^{T-1}\Bigg\{\Bigg[\langle \frac{\partial \mathcal{V}(\Phi_i(\cdot,\psi_T^k);\ \theta)}{\partial\theta}\Big|_{\theta^k}, -\beta^k\Big[\nabla\ell(\hat{\psi}^k(\theta^k))+\xi^k\Big]\rangle \\
&\quad +\frac{\delta\beta_k^2}{2}\|\nabla\ell(\hat{\psi}^k(\theta^k))+\xi^k\|^2\Bigg]\Big[b(\psi_t^k)Y_t(\psi_t^k)+\sigma Z_t(\psi_t^k)\Big]-R_i(\psi_t^k)\Bigg\} \\
&=\frac{1}{N}\sum_{i=1}^{N}\sum_{t=0}^{T-1}\Bigg\{\Bigg[\langle \frac{\partial \mathcal{V}(\Phi_i(\cdot,\psi_T^k);\ \theta)}{\partial\theta}\Big|_{\theta^k}, -\beta^k\Big[\nabla\ell(\hat{\psi}^k(\theta^k))+\xi^k\Big]\rangle+\frac{\delta(\beta^k)^2}{2}\Big(\|\nabla\ell(\hat{\psi}^k(\theta^k))\|^2+\|\xi^k\|^2 \\
&\quad +2\Big\langle\nabla\ell(\hat{\psi}^k(\theta^k)),\xi^k\Big\rangle\Big)\Bigg]\Big[b(\psi_t^k)Y_t(\psi_t^k)+\sigma Z_t(\psi_t^k)\Big]-R_i(\psi_t^k)\Bigg\} \qquad (10)
\end{aligned}
$$

For the second term,

$$
\begin{aligned}
&\mathcal{H}(\cdot,\psi^{k+1};\ \theta^{k+1})-\mathcal{H}(\cdot,\psi^{k};\ \theta^{k+1}) \\
&\leq\langle\nabla\mathcal{H}(\psi^k;\ \theta^{k+1}),\psi^{k+1}-\psi^k\rangle+\frac{L}{2}\|\psi^{k+1}-\psi^k\|^2 \\
&=\langle\nabla\mathcal{H}(\cdot,\psi^k;\ \theta^{k+1}),-\alpha^k[\nabla\mathcal{L}(\psi^k;\ \theta^{k+1})+\upsilon^k]\rangle+\frac{L(\alpha^k)^2}{2}\|\nabla\mathcal{H}(\psi^k;\ \theta^{k+1})+\upsilon^k\|^2 \\
&=-(\alpha^k-\frac{L(\alpha^k)^2}{2})\|\nabla\mathcal{H}(\cdot,\psi^k;\ \theta^{k+1})\|^2+\frac{L(\alpha^k)^2}{2}\|\upsilon^k\|^2-(\alpha_k-L(\alpha^k)^2)\langle\nabla\mathcal{H}(\cdot,\psi^k;\ \theta^{k+1}),\upsilon^k\rangle. \qquad (11)
\end{aligned}
$$

Therefore, we have

$$
\begin{aligned}
&\mathcal{H}(\cdot,\psi^{k+1};\ \theta^{k+2})-\mathcal{H}(\cdot,\psi^{k};\ \theta^{k+1}) \\
&\leq\frac{1}{N}\sum_{i=1}^{N}\Bigg\{\langle\frac{\partial \mathcal{V}(\Phi_i(\cdot,\psi^k);\ \theta)}{\partial\theta}\Big|_{\theta^k},-\beta^k\Big[\nabla\ell(\hat{\psi}^k(\theta^k))+\xi^k\Big]\rangle \\
&\quad+\frac{\delta(\beta^K)^2}{2}(\|\nabla\ell(\hat{\psi}^k(\theta^k))\|^2+\|\xi^k\|^2+2\langle\nabla\ell(\hat{\psi}^k(\theta^k)),\xi^k\rangle)\Bigg\}\Phi_i(\cdot,\psi^k) \\
&\quad-(\alpha^k-\frac{L(\alpha^k)^2}{2})\|\nabla\mathcal{H}(\cdot,\psi^k;\ \theta^{k+1})\|^2+\frac{L(\alpha^k)^2}{2}\|\upsilon^k\|^2-(\alpha^k-L(\alpha^k)^2)\langle\nabla\mathcal{H}(\cdot,\psi^k;\ \theta^{k+1}),\upsilon^k\rangle. \qquad (12)
\end{aligned}
$$

Taking expectation of both sides of Eq.(11) and since $\mathbb{E}[\xi^k]=0,\mathbb{E}[\upsilon^k]=0$, we have

$$
\begin{aligned}
&\mathbb{E}\Big[\mathcal{H}(\cdot,\psi^{k+1};\ \theta^{k+2})\Big]-\mathbb{E}\Big[\mathcal{H}(\cdot,\psi^{k};\ \theta^{k+1})\Big] \\
&\leq\mathbb{E}\frac{1}{N}\sum_{i=1}^{N}\mathcal{H}_i(\cdot,\psi^k)\Bigg\{\langle\frac{\partial \mathcal{V}(\Phi_i(\cdot,\psi^k);\ \theta)}{\partial\theta}\Big|_{\theta^k},-\beta^k\Big[\nabla\ell(\hat{\psi}^k(\theta^k))\Big]\rangle+ \\
&\quad+\frac{\delta(\beta^k)^2}{2}(\|\nabla\ell(\hat{\psi}^k(\theta^k))\|^2+\|\xi^k\|^2)\Bigg\}-\alpha^k\mathbb{E}\Big[\|\nabla\mathcal{H}(\cdot,\psi^k;\ \theta^{k+1})\|^2\Big] \\
&\quad+\frac{L(\alpha^k)^2}{2}\Big\{\mathbb{E}\Big[\|\nabla\mathcal{H}(\cdot,\psi^k;\ \theta^{k+1})\|^2\Big]+\mathbb{E}\Big[\|\upsilon^k\|^2\Big]\Big\}
\end{aligned}
$$

Summing up the above inequalities over $k=1,...,\infty$ in both sides, we obtain

$$
\begin{aligned}
&\sum_{k=1}^{\infty}\alpha^k\mathbb{E}\Big[\|\nabla\mathcal{H}(\cdot,\psi^k;\theta^{k+1})\|^2\Big]+\sum_{k=1}^{\infty}\beta^k\mathbb{E}\frac{1}{N}\sum_{i=1}^{N}\|\mathcal{H}_i(\cdot,\psi^k)\|\|\frac{\partial \mathcal{V}(\Phi_i(\cdot,\psi^k);\ \theta)}{\partial\theta}\Big|_{\theta^k}\|\cdot\|\nabla\ell(\hat{\psi}^k(\theta^k))\| \\
&\leq\sum_{k=1}^{\infty}\frac{L(\alpha^k)^2}{2}\Big\{\mathbb{E}\Big[\|\nabla\mathcal{H}(\cdot,\psi^k;\theta^{k+1})\|^2\Big]+\mathbb{E}\Big[\|\upsilon^k\|^2\Big]\Big\}+\mathbb{E}\Big[\mathcal{H}(\cdot,\psi^1;\ \theta^2)\Big]-\lim_{T\to\infty}\mathbb{E}\Big[\mathcal{H}(\cdot,\psi^{k+1};\theta^{k+2})\Big]
\end{aligned}
$$

$$
\begin{aligned}
&+\sum_{k=1}^{\infty}\frac{\delta(\beta^k)^2}{2}\left\{\frac{1}{N}\sum_{i=1}^{N}\|\mathcal{H}_i(\cdot,\psi^k)\|(\mathbb{E}\|\nabla\ell(\hat{\psi}^k(\theta^k))\|^2+\mathbb{E}\|\xi^k\|^2\right\}\\
\leq&\sum_{k=1}^{\infty}\frac{L(\alpha^k)^2}{2}\{\rho^2+\sigma^2\}+\mathbb{E}\left[\mathcal{H}^{tr}(\cdot,\psi^1;\,\theta^2)\right]+\sum_{k=1}^{\infty}\frac{\delta(\beta^k)^2}{2}\left\{M(\rho^2+\sigma^2)\right\}\leq\infty.
\end{aligned}
$$

The last inequality holds since $\sum_{k=0}^{\infty}(\alpha^k)^2<\infty,\sum_{k=0}^{\infty}(\beta^k)^2<\infty$, and $\frac{1}{N}\sum_{i=1}^{N}\|\mathcal{H}_i(\cdot,\psi^k)\|\leq M$ for limited number of data points' loss is bounded. Thus we have

$$
\sum_{k=1}^{\infty}\alpha^k\mathbb{E}[\|\nabla\mathcal{H}(\cdot,\psi^k;\,\theta^{k+1})\|^2]+\sum_{k=1}^{\infty}\beta^k\mathbb{E}\frac{1}{N}\sum_{i=1}^{n}\|\mathcal{H}_i(\cdot,\psi^k)\|\|\frac{\partial\mathcal{V}(\Phi_i(\cdot,\psi^k);\ \theta)}{\partial\theta}\Big|_{\theta^k}\|\cdot\|\nabla\ell(\hat{\psi}^k(\theta^k))\|\leq\infty. \tag{13}
$$

since

$$
\sum_{k=1}^{\infty}\beta^k\mathbb{E}\frac{1}{N}\sum_{i=1}^{N}\|\mathcal{H}_i(\cdot,\psi^k)\|\|\frac{\partial\mathcal{V}(\Phi_i(\cdot,\psi^k);\ \theta)}{\partial\theta}\Big|_{\theta^k}\|\cdot\|\nabla\ell(\hat{\psi}^k(\theta^k))\|\leq M\delta\rho\sum_{k=1}^{\infty}\beta^k\leq\infty, \tag{14}
$$

This indicates that $\sum_{k=1}^{\infty}\alpha^k\mathbb{E}[\|\nabla\mathcal{H}(\cdot,\psi^k;\ \theta^{k+1})\|^2]<\infty$. Building upon this foundation and referring to Lemma 1, to validate that $\lim_{t\to\infty}\mathbb{E}[\nabla\mathcal{H}(\cdot,\psi^k;\ \theta^{k+1})\|^2]=0$, since $\sum_{k=0}^{\infty}\alpha^k=\infty$, it is essential to demonstrate that

$$
\left|\mathbb{E}[\|\nabla\mathcal{H}(\cdot,\psi^{k+1};\ \theta^{k+2})\|^2]-\mathbb{E}[\|\nabla\mathcal{H}(\cdot,\psi^k;\ \theta^{k+1})\|^2]\right|\leq C\alpha^k, \tag{15}
$$

for some constant C. Based on the inequality

$$
|(\|a\|+\|b\|)(\|a\|-\|b\|)|\leq\|a+b\|\|a-b\|,
$$

we then have

$$
\begin{aligned}
&\left|\mathbb{E}[\|\nabla\mathcal{H}(\cdot,\psi^{k+1};\ \theta^{k+2})\|^2]-\mathbb{E}\left[\|\nabla\mathcal{H}(\cdot,\psi^k;\ \theta^{k+1})\|^2\right]\right|\\
=&\left|\mathbb{E}\left[(\|\nabla\mathcal{H}(\cdot,\psi^{k+1};\ \theta^{k+2})\|+\|\nabla\mathcal{H}(\cdot,\psi^k;\ \theta^{k+1})\|)(\|\nabla\mathcal{H}(\cdot,\psi^{k+1};\ \theta^{k+2})\|-\|\nabla\mathcal{H}(\cdot,\psi^k;\ \theta^{k+1})\|)\right]\right|\\
\leq&\mathbb{E}\left[\left|\|\nabla\mathcal{H}(\cdot,\psi^{k+1};\ \theta^{k+1})\|+\|\nabla\mathcal{H}(\cdot,\psi^k;\ \theta^k)\|)\right|\left|(\|\nabla\mathcal{H}(\cdot,\psi^{k+1};\ \theta^{k+2})\|-\|\nabla\mathcal{H}(\cdot,\psi^k;\ \theta^{k+1})\|)\right|\right]\\
\leq&\mathbb{E}\left[\left\|\nabla\mathcal{H}(\cdot,\psi^{k+1};\ \theta^{k+2})+\nabla\mathcal{H}(\cdot,\psi^k;\ \theta^{k+1})\right\|\left\|\nabla\mathcal{H}(\cdot,\psi^{k+1};\ \theta^{k+2})-\nabla\mathcal{H}(\cdot,\psi^k;\ \theta^{t+1})\right\|\right]\\
\leq&\mathbb{E}\left[(\left\|\nabla\mathcal{H}(\cdot,\psi^{k+1};\ \theta^{k+2})\right\|+\left\|\nabla\mathcal{H}(\cdot,\psi^k;\ \theta^{k+1})\right\|)\left\|\nabla\mathcal{H}(\cdot,\psi^{k+1};\ \theta^{k+2})-\nabla\mathcal{H}(\cdot,\psi^k;\ \theta^{k+1})\right\|\right]\\
\leq&2L\rho\mathbb{E}\left[\|(\psi^{k+1},\theta^{k+2})-(\psi^k,\theta^{k+1})\|\right]\\
\leq&2L\rho\alpha_k\beta_k\mathbb{E}\left[\left\|\left(\nabla\mathcal{H}(\cdot,\psi^k;\ \theta^{k+1})+\upsilon^k,\nabla\ell(\theta^{k+1})+\xi^{k+1}\right)\right\|\right]\\
\leq&2L\rho\alpha_k\beta_k\mathbb{E}\left[\sqrt{\|\nabla\mathcal{H}(\cdot,\psi^k;\ \theta^{k+1})+\upsilon^k\|^2}+\sqrt{\|\nabla\ell(\theta^{k+1})+\xi^{k+1}\|^2}\right]\\
\leq&2L\rho\alpha_k\beta_k\sqrt{\mathbb{E}\left[\|\nabla\mathcal{H}(\cdot,\psi^k;\ \theta^{k+1})+\upsilon^k\|^2\right]+\mathbb{E}\left[\|\nabla\ell(\theta^{k+1})+\xi^{k+1}\|^2\right]}\\
\leq&2L\rho\alpha_k\beta_k\sqrt{\mathbb{E}\left[\|\nabla\mathcal{H}(\cdot,\psi^k;\ \theta^{k+1})\|^2\right]+\mathbb{E}\left[\|\upsilon^k\|^2\right]+\mathbb{E}\left[\|\xi^{k+1}\|^2\right]+\mathbb{E}\left[\|\nabla\ell(\theta^{k+1})\|^2\right]}\\
\leq&2L\rho\alpha^k\beta^k\sqrt{2\sigma^2+2\rho^2}\\
\leq&2\sqrt{2(\sigma^2+\rho^2)}L\rho\beta^1\alpha^k.
\end{aligned} \tag{16}
$$

According to the above inequality, we can conclude that our algorithm can achieve

$$
\lim_{k\to\infty}\mathbb{E}\left[\|\nabla\mathcal{H}(\cdot,\psi^k;\ \theta^{k+1})\|^2\right]=0. \tag{17}
$$

The proof is completed. □

6.2 Convergence proof of the meta loss

Assume that we have a small amount of meta dataset with M data points $\{(x_i',y_i'),1\leq i\leq M\}$ with clean labels, and the meta loss is

$$
\ell(\psi(\theta))=\frac{1}{M}\sum_{i=1}^{M}\ell_i(\psi(\theta)), \tag{18}
$$

Let's suppose we have another N training data points, $\{(x_i, y_i), 1 \leq i \leq N\}$, where $M \ll N$, and the training loss is

$$\mathcal{L}(\psi;\ \theta) = \frac{1}{N}\sum_{i=1}^{N}\left[\sum_{t=0}^{T-1} R_i(t, X_{i,t}, \mu_t, \psi_{i,t}) + \mathcal{V}(\Phi_i(\cdot, \psi_T);\ \theta)\Phi_i(X_{i,T}, \mu_T, \psi_T)\right]. \tag{19}$$

Lemma 2 Suppose a meta loss function ℓ that is Lipschitz smooth with a constant L, and a function $\mathcal{V}(\cdot)$ that is differentiable, with a δ-bounded gradient, and twice differentiable with its Hessian bounded by $\mathcal{B}$. If the loss function's gradients are bounded by ρ to metadata points, then the gradients of θ to ℓ are Lipschitz continuous.

Proof The gradient of the parameter θ concerning the meta loss ℓ_i can be expressed as

$$\nabla_\theta \ell_i(\hat{\psi}^k(\theta))\Big|_{\theta^k} = -\frac{\alpha}{M}\sum_{j=1}^{M}\left(\frac{\partial \ell_i(\hat{\psi})}{\partial \hat{\psi}}\Big|_{\hat{\psi}^k}^{T}\frac{\partial \Phi_j(\cdot, \psi)}{\partial \psi}\Big|_{\psi^k}\right)\frac{\partial \mathcal{V}(\Phi_j(\cdot, \psi^k);\ \theta)}{\partial \theta}\Big|_{\theta^k}, \tag{20}$$

Let $\mathcal{V}_j(\cdot;\ \theta) = \mathcal{V}(\Phi_j(\cdot, \psi^k);\ \theta)$ and G_{ij} being defined in Eq.(39). Taking gradient of θ in both sides of Eq.(20), we have

$$\nabla^2_{\theta^2}\ell_i(\hat{\psi}^k(\theta))\Big|_{\theta^k} = -\frac{\alpha}{N}\sum_{j=1}^{n}\left[\frac{\partial}{\partial \theta}\left(G_{ij}\right)\Big|_{\theta^k}\frac{\partial \mathcal{V}_j(\cdot;\ \theta)}{\partial \theta}\Big|_{\theta^k} + \left(G_{ij}\right)\frac{\partial^2 \mathcal{V}_j(\cdot;\ \theta)}{\partial \theta^2}\Big|_{\theta^k}\right].$$

For the first term on the right-hand side, we have that

$$\begin{aligned}
&\left\|\frac{\partial}{\partial \theta}(G_{ij})\Big|_{\theta^k}\frac{\partial \mathcal{V}_j(\cdot;\ \theta)}{\partial \theta}\Big|_{\theta^k}\right\| \\
\leq&\delta\left\|\frac{\partial}{\partial \hat{\psi}}\left(\frac{\partial \ell_i(\hat{\psi})}{\partial \theta}\Big|_{\theta^k}\right)\Big|_{\hat{\psi}^k}^{T}\frac{\partial \Phi_j(\cdot, \psi)}{\partial \psi}\Big|_{\psi^k}\right\| \\
=&\delta\left\|\frac{\partial}{\partial \hat{\psi}}\left(\frac{\partial \ell_i(\hat{\psi})}{\partial \hat{\psi}}\Big|_{\hat{\psi}^k}\frac{-\alpha}{N}\sum_{m=1}^{n}\frac{\partial \Phi_m(\cdot, \psi)}{\partial \psi}\Big|_{\psi^k}\frac{\partial \mathcal{V}_k(\cdot;\ \theta)}{\partial \theta}\Big|_{\theta^k}\right)\Big|_{\hat{\psi}^k}^{T}\frac{\partial \Phi_j(\cdot, \psi)}{\partial \psi}\Big|_{\psi^k}\right\| \\
=&\delta\left\|\left(\frac{\partial^2 \ell_i(\hat{\psi})}{\partial \hat{\psi}^2}\Big|_{\hat{\psi}^k}\frac{-\alpha}{N}\sum_{m=1}^{n}\frac{\partial \Phi_m(\cdot, \psi)}{\partial \psi}\Big|_{\psi^k}\frac{\partial \mathcal{V}_m(\cdot;\ \theta)}{\partial \theta}\Big|_{\theta^k}\right)\Big|_{\hat{\psi}^k}^{T}\frac{\partial \Phi_j(\cdot, \psi)}{\partial \psi}\right\| \leq \alpha L \rho^2 \delta^2,
\end{aligned} \tag{21}$$

since $\left\|\frac{\partial^2 \ell_i(\hat{\psi})}{\partial \hat{\psi}^2}\Big|_{\hat{\psi}^k}\right\| \leq L$, $\left\|\frac{\partial \Phi_j(\cdot,\psi)}{\partial \psi}\Big|_{\psi^k}\right\| \leq \rho$, $\left\|\frac{\partial \mathcal{V}_j(\cdot;\ \theta)}{\partial \theta}\Big|_{\theta^k}\right\| \leq \delta$. For the second term, we have

$$\left\|(G_{ij})\frac{\partial^2 \mathcal{V}_j(\cdot;\ \theta)}{\partial \theta^2}\Big|_{\theta^k}\right\| = \left\|\frac{\partial \ell_i(\hat{\psi})}{\partial \hat{\psi}}\Big|_{\hat{\psi}^k}^{T}\frac{\partial \Phi_j(\cdot, \psi)}{\partial \psi}\Big|_{\psi^k}\frac{\partial^2 \mathcal{V}_j(\cdot;\ \theta)}{\partial \theta^2}\Big|_{\theta^k}\right\| \leq \mathcal{B}\rho^2, \tag{22}$$

since $\left\|\frac{\partial \ell_i(\hat{\psi})}{\partial \hat{\psi}}\Big|_{\hat{\psi}^k}^{T}\right\| \leq \rho$, $\left\|\frac{\partial^2 \mathcal{V}_j(\cdot;\ \theta)}{\partial \theta^2}\Big|_{\theta^k}\right\| \leq \mathcal{B}$. Combining the above two inequalities Eq.(21) and (22), we have

$$\left\|\nabla^2_{\theta^2}\ell_i(\hat{\psi}^k(\theta))\Big|_{\theta^k}\right\| \leq \alpha\rho^2(\alpha L\delta^2 + \mathcal{B}). \tag{23}$$

Define $L_V = \alpha\rho^2(\alpha L\delta^2 + \mathcal{B})$, based on Lagrange mean value theorem, we have

$$\|\nabla \ell(\hat{\psi}^k(\theta_1)) - \nabla \ell(\hat{\psi}^k(\theta_2))\| \leq L_V\|\theta_1 - \theta_2\|,\ for\ \forall\ \theta_1, \theta_2, \tag{24}$$

where $\nabla \ell(\hat{\psi}^k(\theta_1)) = \nabla_\theta \ell_i(\hat{\psi}^k(\theta))\Big|_{\theta_1}$. □

Theorem 2 Assume the meta loss function ℓ is Lipschitz smooth with constant L, and $\mathcal{V}(\cdot)$ is differential with a δ-bounded gradient and twice differential with its Hessian bounded by $\mathcal{B}$, and ℓ have ρ-bounded gradients concerning metadata points. Let the learning rate α^k satisfies $\alpha^k = \min\{1, \frac{k}{T}\}$, for some $k > 0$, such that $\frac{k}{T} < 1$, and $\beta^k, 1 \leq k \leq K$ is a monotone descent sequence, $\beta^1 = \min\{\frac{1}{L}, \frac{c}{\sigma\sqrt{T}}\}$ for some $c > 0$, such that $\frac{\sigma\sqrt{T}}{c} \geq L$ and $\sum_{t=1}^{\infty}\beta_t \leq \infty, \sum_{t=1}^{\infty}\beta_t^2 \leq \infty$. Then meta network can achieve $\mathbb{E}[\|\nabla \ell(\hat{\psi}^k(\theta^k))\|^2] \leq \epsilon$ in $\mathcal{O}(1/\epsilon^2)$ steps. More specifically,

$$\min_{0\leq k\leq K}\mathbb{E}[\|\nabla \ell(\hat{\psi}^k(\theta^k))\|^2] \leq \mathcal{O}(\frac{C}{\sqrt{T}}), \tag{25}$$

where C is some constant independent of the convergence process, σ is the variance of randomly drawing uniformly mini-batch data points.

Proof The update of θ in the k-th iteration is as follows

$$\theta^{k+1}=\theta^{k}-\frac{\beta}{M}\sum_{i=1}^{M}\nabla_{\theta}\ell_i(\hat{\psi}^{k}(\theta))\Big|_{\theta^{k}}. \tag{26}$$

This can be written as:

$$\theta^{k+1}=\theta^{k}-\beta^{k}\nabla\ell(\hat{\psi}^{k}(\theta^{k}))\big|_{\Xi^{k}}. \tag{27}$$

Since the mini-batch Ξ^{k} is drawn uniformly from the entire dataset, we can rewrite the update equation as

$$\theta^{k+1}=\theta^{k}-\beta^{k}[\nabla\ell(\hat{\psi}^{k}(\theta^{k}))+\xi^{k}], \tag{28}$$

where $\xi^{k}=\nabla\ell(\hat{\psi}^{k}(\theta^{k}))\big|_{\Xi^{k}}-\nabla\ell(\hat{\psi}^{k}(\theta^{k}))$. The random variables ξ^{k} have finite variance because the samples Ξ^{k} are drawn independently and identically distributed (i.i.d.) from a finite sample set. Additionally, the expected value $\mathbb{E}[\xi^{k}]=0$, as the samples are drawn uniformly at random. Observe that

$$\begin{aligned}&\ell(\hat{\psi}^{k+1}(\theta^{k+1}))-\ell(\hat{\psi}^{k}(\theta^{k}))\\=&\left\{\ell(\hat{\psi}^{k+1}(\theta^{k+1}))-\ell(\hat{\psi}^{k}(\theta^{k+1}))\right\}+\left\{\ell(\hat{\psi}^{k}(\theta^{k+1}))-\ell(\hat{\psi}^{k}(\theta^{k}))\right\}.\end{aligned} \tag{29}$$

By Lipschitz smoothness of meta loss function ℓ, we have

$$\begin{aligned}&\ell(\hat{\psi}^{k+1}(\theta^{k+1}))-\ell(\hat{\psi}^{k}(\theta^{k+1}))\\\leq&\langle\nabla\ell(\hat{\psi}^{k}(\theta^{k+1})),\hat{\psi}^{k+1}(\theta^{k+1})-\hat{\psi}^{k}(\theta^{k+1})\rangle+\frac{L}{2}\|\hat{\psi}^{k+1}(\theta^{k+1})-\hat{\psi}^{k}(\theta^{k+1})\|^{2}\end{aligned}$$

since $\hat{\psi}^{k+1}(\theta^{k+1})-\hat{\psi}^{k}(\theta^{k+1})=-\frac{\alpha^{k}}{N}\sum_{i=1}^{n}\mathcal{V}(\Phi_i(\cdot,\psi^{k+1});\ \theta^{k+1})\nabla_{\psi}\Phi_i(\cdot,\psi)\Big|_{\psi^{k+1}}$, we have

$$\|\ell(\hat{\psi}^{k+1}(\theta^{k+1}))-\ell(\hat{\psi}^{k}(\theta^{k+1}))\|\leq\alpha^{k}\rho^{2}+\frac{L(\alpha^{k})^{2}}{2}\rho^{2}=\alpha^{k}\rho^{2}(1+\frac{\alpha^{k}L}{2}) \tag{30}$$

where $\left\|\frac{\partial\Phi_j(\cdot,\psi)}{\partial\psi}\Big|_{\psi^{k}}\right\|\leq\rho$, $\left\|\frac{\partial\ell_i(\hat{\psi})}{\partial\hat{\psi}}\Big|_{\hat{\psi}^{k}}^{T}\right\|\leq\rho$.

By Lipschitz continuity of $\nabla\ell(\hat{\psi}^{k}(\theta))$ according to Lemma 2, we can obtain the following

$$\begin{aligned}&\ell(\hat{\psi}^{k}(\theta^{k+1}))-\ell(\hat{\psi}^{k}(\theta^{k}))\\\leq&\langle\nabla\ell(\hat{\psi}^{k}(\theta^{k})),\theta^{k+1}-\theta^{k}\rangle+\frac{L}{2}\|\theta^{k+1}-\theta^{k}\|^{2}\\=&\langle\nabla\ell(\hat{\psi}^{k}(\theta^{k})),-\beta^{k}[\nabla\ell(\hat{\psi}^{k}(\theta^{k}))+\xi^{k}]\rangle+\frac{L(\beta^{k})^{2}}{2}\|\nabla\ell(\hat{\psi}^{k}(\theta^{k}))+\xi^{k}\|^{2}\\=&-(\beta^{k}-\frac{L(\beta^{k})^{2}}{2})\|\nabla\ell(\hat{\psi}^{k}(\theta^{k}))\|^{2}+\frac{L(\beta^{k})^{2}}{2}\|\xi^{k}\|^{2}-(\beta^{k}-L(\beta^{k})^{2})\langle\nabla\ell(\hat{\psi}^{k}(\theta^{k})),\xi^{k}\rangle.\end{aligned} \tag{31}$$

Thus Eq.(29) satisfies

$$\begin{aligned}\ell(\hat{\psi}^{k+1}(\theta^{k+1}))-\ell(\hat{\psi}^{k}(\theta^{k}))\leq&\alpha^{k}\rho^{2}(1+\frac{\alpha^{k}L}{2})-(\beta^{k}-\frac{L(\beta^{k})^{2}}{2})\|\nabla\ell(\hat{\psi}^{k}(\theta^{k}))\|^{2}\\&+\frac{L(\beta^{k})^{2}}{2}\|\xi^{k}\|^{2}-(\beta^{k}-L(\beta^{k})^{2})\langle\nabla\ell(\hat{\psi}^{k}(\theta^{k})),\xi^{k}\rangle.\end{aligned} \tag{32}$$

Rearranging the terms, we can obtain

$$\begin{aligned}(\beta^{k}-\frac{L(\beta^{k})^{2}}{2})\|\nabla\ell(\hat{\psi}^{k}(\theta^{k}))\|^{2}\leq&\alpha^{k}\rho^{2}(1+\frac{\alpha^{k}L}{2})+\ell(\hat{\psi}^{k}(\theta^{k}))-\ell(\hat{\psi}^{k+1}(\theta^{k+1}))\\&+\frac{L(\beta^{k})^{2}}{2}\|\xi^{k}\|^{2}-(\beta^{k}-L(\beta^{k})^{2})\langle\nabla\ell(\hat{\psi}^{k}(\theta^{k})),\xi^{k}\rangle.\end{aligned} \tag{33}$$

Summing up the above inequalities and rearranging the terms, we can obtain

$$\begin{aligned}&\textstyle\sum_{k=1}^{K}(\beta^{k}-\frac{L(\beta^{k})^{2}}{2})\|\nabla\ell(\hat{\psi}^{k}(\theta^{k}))\|^{2}\\&\leq\ell(\hat{\psi}^{1})(\theta^{1})-\ell(\hat{\psi}^{K+1}(\theta^{K+1}))+\sum_{k=1}^{K}\alpha^{k}\rho^{2}(1+\frac{\alpha^{k}L}{2})\\&-\sum_{k=1}^{K}(\beta^{k}-L(\beta^{k})^{2})\langle\nabla\ell(\hat{\psi}^{k}(\theta^{k})),\xi^{k}\rangle+\frac{L}{2}\sum_{k=1}^{K}(\beta_{k})^{2}\|\xi^{k}\|^{2}\end{aligned}$$

$$\leq \ell(\hat{\psi}^1(\theta^1)) + \sum_{k=1}^{K} \alpha^k \rho^2 (1 + \frac{\alpha^k L}{2}) - \sum_{k=1}^{K} (\beta^k - L(\beta^k)^2) \langle \nabla \ell(\hat{\psi}^k(\theta^k)), \xi^k \rangle + \frac{L}{2} \sum_{k=1}^{K} (\beta^k)^2 \|\xi^k\|^2, \tag{34}$$

By taking expectations with respect to ξ^K on both sides of Eq.(34), we obtain

$$\sum_{k=1}^{K} (\beta^k - \frac{L(\beta^k)^2}{2}) \mathbb{E}_{\xi^K} \|\nabla \ell(\hat{\psi}^k(\theta^k))\|^2 \leq \ell(\hat{\psi}^1(\theta^1)) + \sum_{k=1}^{K} \alpha^k \rho^2 (1 + \frac{\alpha^k L}{2}) + \frac{L\sigma^2}{2} \sum_{k=1}^{K} (\beta^k)^2, \tag{35}$$

since $\mathbb{E}_{\xi^K} \langle \nabla \ell(\theta^k), \xi^k \rangle = 0$ and $\mathbb{E}[\|\xi^k\|^2] \leq \sigma^2$, where σ^2 denotes the variance of ξ^k. Consequently, we can further deduce that

$$\begin{aligned}
&\min_k \mathbb{E}\left[\|\nabla \ell(\hat{\psi}^k(\theta^k))\|^2\right] \\
&\leq \frac{\sum_{k=1}^{K} (\beta^k - \frac{L(\beta^k)^2}{2}) \mathbb{E}_{\xi^K} \|\nabla \ell(\hat{\psi}^k(\theta^k))\|^2}{\sum_{k=1}^{K} (\beta^k - \frac{L(\beta^k)^2}{2})} \\
&\leq \frac{1}{\sum_{k=1}^{K} (2\beta^k - L(\beta^k)^2)} \left[2\ell(\hat{\psi}^1(\theta^1)) + \sum_{k=1}^{K} \alpha^k \rho^2 (2 + \alpha^k L) + L\sigma^2 \sum_{k=1}^{K} (\beta^k)^2 \right] \\
&\leq \frac{1}{\sum_{k=1}^{K} \beta^k} \left[2\ell(\hat{\psi}^1(\theta^1)) + \sum_{k=1}^{K} \alpha^k \rho^2 (2 + \alpha^k L) + L\sigma^2 \sum_{k=1}^{K} (\beta^k)^2 \right] \\
&\leq \frac{1}{K\beta^k} \left[2\ell(\hat{\psi}^1(\theta^1)) + \alpha^1 \rho^2 T (2 + L) + L\sigma^2 \sum_{k=1}^{K} (\beta^k)^2 \right] \\
&= \frac{2\ell(\hat{\psi}^1(\theta^1))}{K} \frac{1}{\beta^k} + \frac{2\alpha^1 \rho^2 (2 + L)}{\beta^k} + \frac{L\sigma^2}{K} \sum_{k=1}^{K} \beta^k \\
&\leq \frac{2\ell(\hat{\psi}^1(\theta^1))}{K} \frac{1}{\beta^k} + \frac{2\alpha^1 \rho^2 (2 + L)}{\beta^k} + L\sigma^2 \beta^k \\
&= \frac{\ell(\hat{\psi}^1(\theta^1))}{K} \max\{L, \frac{\sigma\sqrt{K}}{c}\} + \min\{1, \frac{k}{K}\} \max\{L, \frac{\sigma\sqrt{T}}{c}\} \rho^2 (2 + L) + L\sigma^2 \min\{\frac{1}{L}, \frac{c}{\sigma\sqrt{T}}\} \\
&\leq \frac{\sigma\ell(\hat{\psi}^1(\theta^1)}{c\sqrt{K}} + \frac{k\sigma\rho^2 (2 + L)}{c\sqrt{K}} + \frac{L\sigma c}{\sqrt{K}} \\
&= \mathcal{O}(\frac{1}{\sqrt{K}}).
\end{aligned} \tag{36}$$

The third inequality holds due to $\sum_{k=1}^{K} (2\beta^k - L(\beta^k)^2) \geq \sum_{k=1}^{K} \beta^k$. As a result, it is demonstrated that this approach consistently achieves $\min_{0 \leq k \leq K} \mathbb{E}[\|\nabla \ell(\theta^k)\|^2] \leq \mathcal{O}(\frac{1}{\sqrt{K}})$ within K iterations. This concludes the proof of Theorem 2. □

6.3 Derivation of the update details for meta parameters

We now recall the updated equation for the parameters of the data re-weighting strategy, given by

$$\theta^{k+1} = \theta^k - \beta \frac{1}{M} \sum_{i=1}^{M} \nabla_\theta \ell_i(\hat{\psi}^k(\theta)) \Big|_{\theta^k}, \tag{37}$$

The computation of Eq. (37) through backpropagation is detailed in the following derivation, as

$$\begin{aligned}
&\frac{1}{M} \sum_{i=1}^{M} \nabla_\theta \ell_i(\hat{\psi}^k(\theta)) \Big|_{\theta^k} \\
=&\frac{1}{M} \sum_{i=1}^{M} \frac{\partial \ell_i(\hat{\psi})}{\partial \hat{\psi}(\theta)} \Big|_{\hat{\psi}^k} \sum_{j=1}^{n} \frac{\partial \hat{\psi}^k(\theta)}{\partial \mathcal{V}(\Phi_j(\cdot, \psi^k);\ \theta)} \frac{\partial \mathcal{V}(\Phi_j(\cdot, \psi^k);\ \theta)}{\partial \theta} \Big|_{\theta^k} \\
=&- \frac{\alpha}{NM} \sum_{i=1}^{M} \frac{\partial \ell_i(\hat{\psi})}{\partial \hat{\psi}} \Big|_{\hat{\psi}^k} \sum_{j=1}^{N} \frac{\partial \Phi_j(\cdot, \psi)}{\partial \psi} \Big|_{\psi^k} \frac{\partial \mathcal{V}(\Phi_j(\cdot, \psi^k);\ \theta)}{\partial \theta} \Big|_{\theta^k}
\end{aligned}$$

$$= -\frac{\alpha}{N}\sum_{j=1}^{N}\left(\frac{1}{M}\sum_{i=1}^{M}\frac{\partial \ell_i(\hat{\psi})}{\partial \hat{\psi}}\Big|_{\hat{\psi}^k}^{T}\frac{\partial \Phi_j(\cdot,\psi)}{\partial \psi}\Big|_{\psi^k}\right)\frac{\partial \mathcal{V}(\Phi_j(\cdot,\psi^k);\ \theta)}{\partial \theta}\Big|_{\theta^k}. \tag{38}$$

Let

$$G_{ij} = \frac{\partial \ell_i(\hat{\psi})}{\partial \hat{\psi}}\Big|_{\hat{\psi}^k}^{T}\frac{\partial \Phi_j(\cdot,\psi)}{\partial \psi}\Big|_{\psi^k}, \tag{39}$$

by substituting Eq.(38) into Eq.(37), we can get:

$$\theta^{k+1} = \theta^k + \frac{\alpha\beta}{N}\sum_{j=1}^{N}\left(\frac{1}{M}\sum_{i=1}^{M}G_{ij}\right)\frac{\partial \mathcal{V}(\Phi_j(\cdot,\psi^k);\ \theta)}{\partial \theta}\Big|_{\theta^k}. \tag{40}$$

7 Detailed description of the weighted mean-field MSA

7.1 Weighted mean-field MSA

In this section, we illustrate the iterative update process of the weighted mean-field MSA in Alg.S2, comprising the state equation, the costate equation, and the maximization of the Hamiltonian.

Algorithm S2 Weighted Mean-field MSA

```
Initialize $\boldsymbol{\psi}^0 = \{\psi_t^0 \in \Psi : t = 0\dots,T-1\}$.
for $k = 0$ to $K$ do
    Solve the forward SDE:
        $\mathrm{d}X_t^k = \left[a(\mu_t - X_t^k) + \psi_t^k\right]\mathrm{d}t + \sigma \mathrm{d}W_t, \quad X_0^k = x,$
    Solve the backward SDE:
        $\mathrm{d}Y_t^k = -\nabla_x \mathcal{H}(t, X_t^k, Y_t^k, \mu_t^k, \psi_t^k)\mathrm{d}t + Z_t^k \mathrm{d}W_t, \quad Y_T^k = -\mathcal{V}(\Phi(\cdot);\ \theta)\nabla_x \Phi(X_T^k, \mu_T^k, \psi_T^k),$
    For each $t \in [0, T-1]$, update the state control $\psi_t^{k+1}$:
        $\psi_t^{k+1} = \arg\max_{\psi \in \Psi} \mathcal{H}(t, X_t^k, Y_t^k, Z_t^k, \mu_t, \psi).$
end for
```

7.2 Error estimate for the weighted mean-field MSA

In this section, we derive a rigorous error estimate for the weighted mean-field MSA, aiding in comprehending its stochastic dynamic. The discrete-time analogue of the weighted stochastic control problem is

$$\begin{aligned}&\min_{\psi,\theta}\frac{1}{N}\sum_{i=1}^{N}\left[\sum_{t=0}^{T-1}R_i(t, X_{i,t}, \mu_t, \psi_{i,t}) + \mathcal{V}(\Phi_i(X_{i,T}, \mu_T, \psi_{i,T});\ \theta)\Phi_i(X_{i,T}, \mu_T, \psi_{i,T})\right],\\ &\text{s.t.} X_{i,t+1} = X_{i,t} + \left[a(\mu_t - X_{i,t}) + \psi_{i,t}\right]\Delta t + \sigma \Delta W.\end{aligned} \tag{41}$$

Let us now make the following assumptions:

Assumption 1 The terminal cost function Φ in Eq.(41) is twice continuously differentiable, with Φ and $\nabla\Phi$ satisfying a Lipschitz condition. There is a constant $K \geq 0$ such that $\forall X_T, X_T' \in \mathbb{R}, \forall \psi_T \in \Psi$

$$|\Phi(\cdot, X_T) - \Phi(\cdot, X_T')| + \|\nabla\Phi(\cdot, X_T) - \nabla\Phi(\cdot, X_T')\| \leq K\left\|\frac{X_T - X_T'}{\mathcal{V}(\Phi(\cdot);\ \theta)}\right\|.$$

Assumption 2 From Eq. (41), the running cost function R and the drift function b are jointly continuous in t and twice continuously differentiable in X_t. The functions b, $\nabla_x b$, R, and $\nabla_x R$ satisfy Lipschitz conditions in X_t, uniformly in t and ψ_t. Given that σ is constant, it does not influence the error estimate. There exists a constant $K \geq 0$ such that $\forall x \in \mathbb{R}^d$, $\forall \psi_t \in \Psi$, and $\forall t \in [0, T-1]$

$$\begin{aligned}\|b(X_t,\cdot) - b(X_t',\cdot)\| + \|\nabla_x b(X_t,\cdot) - \nabla_x b(X_t',\cdot)\| + |R(X_t,\cdot) - R(X_t',\cdot)| \\ + \|\nabla_x R(X_t,\cdot) - \nabla_x R(X_t',\cdot)\| \leq K\|X_t - X_t'\|\end{aligned}$$

With these assumptions, we provide the following error estimate for weighted mean-field MSA.

Lemma 3 Suppose Assumptions 1 and 2 hold. Then for arbitrary admissible controls ψ and ψ' there exists a constant $C > 0$ such that

$$\begin{aligned}\mathcal{L}(\psi) - \mathcal{L}(\psi') \leq & -\sum_{i=1}^{N}\sum_{t=0}^{T-1}\left[\mathcal{H}(t, X_{i,t}, Y_{i,t}, P_{i,t}, \psi_{i,t}) - \mathcal{H}(t, X_{i,t}, Y_{i,t}, P_{i,t}, \psi_{i,t}')\right] \\ & + \frac{C}{N}\sum_{i=1}^{N}\sum_{t=0}^{T-1}\|b(X_{i,t}, Y_{i,t}, P_{i,t}, \psi_{i,t}) - b(X_{i,t}, Y_{i,t}, P_{i,t}, \psi_{i,t}')\|^2 \\ & + \frac{C}{N}\sum_{i=1}^{N}\sum_{t=0}^{T-1}\|\nabla_x b(X_{i,t}, Y_{i,t}, P_{i,t}, \psi_{i,t}) - \nabla_x b(X_{i,t}, Y_{i,t}, P_{i,t}, \psi_{i,t}')\|^2, \\ & + \frac{C}{N}\sum_{i=1}^{N}\sum_{t=0}^{T-1}\|\nabla_x R(X_{i,t}, Y_{i,t}, P_{i,t}, \psi_{i,t}) - \nabla_x R(X_{i,t}, Y_{i,t}, P_{i,t}, \psi_{i,t}')\|^2, \end{aligned} \tag{42}$$

The proof follows a discrete Gronwall's lemma.

Lemma 4 Let $K \geq 0$, and let u_t and w_t be non-negative, real-valued sequences such that

$$u_{t+1} \leq K u_t + w_t,$$

for $t = 0, \ldots, T-1$. Then, for every $t = 0, \ldots, T$, the inequality

$$u_t \leq \max(1, K^T)\left(u_0 + \sum_{s=0}^{T-1} w_s\right)$$

holds.

Proof We prove by induction the following inequality:

$$u_t \leq \max(1, K^t)\left(u_0 + \sum_{s=0}^{T-1} w_s\right),$$

from which the lemma immediately follows. The base case $t = 0$ is trivial.

Assuming the inequality holds for some t, we have

$$\begin{aligned} u_{t+1} &\leq K u_t + w_t, \\ &\leq K \cdot \max(1, K^t)\left(u_0 + \sum_{s=0}^{T-1} w_s\right) + w_t, \\ &= \max(1, K^{t+1}) \cdot \frac{K}{\max(1, K^t)}\left(u_0 + \sum_{s=0}^{T-1} w_s\right) + \max(1, K^{t+1}) \cdot \frac{w_t}{\max(1, K^{t+1})}, \\ &= \max(1, K^{t+1})\left(\frac{K}{\max(1, K^t)}\left(u_0 + \sum_{s=0}^{T-1} w_s\right) + \frac{w_t}{\max(1, K^{t+1})}\right). \end{aligned}$$

Now, consider the expression $\frac{K}{\max(1,K^t)}$

$$\frac{K}{\max(1, K^t)} = \begin{cases} K, & \text{if } K \leq 1, \\ K^{1-t}, & \text{if } K > 1, \end{cases}$$

When $K \leq 1$, we have

$$u_{t+1} \leq \max(1, K^{t+1}) \left(K \left(u_0 + \sum_{s=0}^{T-1} w_s \right) + w_t \right),$$
$$\leq \left(u_0 + \sum_{s=0}^{t} w_s \right),$$

since $K \leq 1$ implies $K^{t+1} \leq 1$.

When $K > 1$, we have

$$\frac{K}{K^t} = K^{1-t},$$

and therefore

$$u_{t+1} \leq K^{t+1} \left(u_0 + \sum_{s=0}^{T-1} w_s \right) + w_t,$$
$$u_{t+1} \leq \max(1, K^{t+1}) \left(u_0 + \sum_{s=0}^{t} w_s \right).$$

Thus, the inequality (7.2) holds for $t+1$. By mathematical induction, the inequality holds for all $t = 0, \ldots, T$, proving this lemma.

This proves (7.2) and hence the lemma. □

We begin by proving a preliminary lemma that provides an estimate for the magnitude of Y_t for any arbitrary $\psi \in \Psi$. Throughout this proof, C will denote a generic constant that does not depend on ψ, ψ', and the batch size S, but may depend on other fixed parameters such as T and the Lipschitz constants K specified in Assumptions 1–2. The value of C may vary from line to line, while retaining the same dependencies, to minimize notational complexity.

Lemma 5 There exists a constant $C > 0$ such that, for each $t = 0, \ldots, T$ and for every $\psi \in \Psi$, the following holds:

$$\|Y_{i,t}\| \leq \frac{C}{N},$$

for all $i = 1, \ldots, N$.

Proof First, observe that $Y_{i,t} = -\frac{1}{N} \nabla \mathcal{V}(\Phi(\cdot);\ \cdot)\Phi(X_{i,t})$ and $\|Y_{i,t}\| = \frac{1}{N} \|\nabla \mathcal{V}(\Phi(\cdot);\ \cdot)\Phi(X_{i,t})\|$.

By Assumption 1, it follows that

$$\|Y_{i,t}\| \leq \frac{K}{N}.$$

where the constant K absorbs the effect of $\mathcal{V}(\Phi(\cdot);\ \theta)$.

For each $0 \leq t < T$,

$$\|Y_{i,t}\| = \|\nabla_x \mathcal{H}(t, X_{i,t}, Y_{i,t+1}, Z_{i,t}, \mu_t, \psi'_{i,t})\|,$$
$$= \|\nabla_x b(t, X_{i,t}, \mu_t, \psi'_{i,t})^T Y_{i,t+1} + \frac{1}{N} \nabla_x R(t, X_{i,t}, \mu_t, \psi_{i,t})\|,$$
$$\leq \|\nabla_x b(t, X_{i,t}, \mu_t, \psi'_{i,t})^T\| \cdot \|Y_{i,t+1}\| + \frac{1}{N} \|\nabla_x R(t, X_{i,t}, \mu_t, \psi_{i,t})\|,$$
$$\leq K\|Y_{i,t+1}\| + \frac{K}{N}.$$

Define $u_t = \|Y_{i,t}\|$, $w_t = \frac{K}{N}$, and K is a positive constant.

Then

$$u_t \leq K u_{t+1} + w_t.$$

By applying Gronwall's inequality (Lemma 4) in reverse time

$$u_t \leq \max(1, K^T) \left(u_T + \sum_{s=t}^{T-1} w_s \right),$$

Since $u_T = \frac{K}{N}$,

$$u_t \leq \max(1, K^T) \left(\frac{K}{N} + \frac{T-t}{N} K \right),$$

This simplifies to

$$u_t \leq \frac{C}{N},$$

where $C = K \max(1, K^T)(1+T)$. □

We are now ready to prove Theorem 3.

Proof Recall the Hamiltonian definition

$$\mathcal{H}(t, X_t, Y_t, Z_t, \mu_t, \psi_t) = [a(\mu_t - X_t) + \psi_t] \cdot Y_t + \sigma^\top Z_t - R(t, X_t, \mu_t, \psi_t).$$

Let us define the quantity

$$I(X_t, Y_t, \psi_t) := \sum_{t=0}^{T-1} \Big[Y_{t+1} \cdot X_{t+1} - \mathcal{H}(t, X_t, Y_{t+1}, Z_t, \mu_t, \psi_t) - \sigma^\top Z_t - R(t, X_t, \mu_t, \psi_t) \Big]$$

Next, we consider the linearized form given by

$$\begin{aligned} b(t, X_{t+1}, \mu_{t+1}, \psi_{t+1}) &= b(t, X_t, \mu_t, \psi_t) + \nabla_x b(t, X_t, \mu_t, \psi_t)(\psi_t - X_t) \\ &= [a(\mu_t - X_t) + \psi_t] + \nabla_x [a(\mu_t - X_t) + \psi_t] (\psi_t - X_t), \end{aligned} \tag{43}$$

From Eq.(43), we have $I(X_t, Y_t, \psi_t) = 0$ for $\psi \in \Psi$. Now, fixing a specific sample i, we obtain the following estimates

$$\begin{aligned} 0 &= I(X_t, Y_t, \psi_t) - I(X'_t, Y'_t, \psi'_t) \\ &= \sum_{t=0}^{T-1} \Big[\big(Y_{t+1} \cdot X_{t+1} - Y'_{t+1} \cdot X'_{t+1}\big) - \big(\mathcal{H}(t, X_t, Y_{t+1}, Z_t, \mu_t, \psi_t) - \mathcal{H}(t, X'_t, Y'_{t+1}, Z'_t, \mu'_t, \psi'_t)\big) \\ &\quad - \Big(\sigma^\top Z_t - \sigma^\top Z'_t\Big) - \big(R(t, X_t, \mu_t, \psi_t) - R(t, X'_t, \mu'_t, \psi'_t)\big) \Big] \\ &= \sum_{t=0}^{T-1} \Big[(Y_{t+1} - Y'_{t+1}) \cdot X_{t+1} + Y'_{t+1} \cdot (X_{t+1} - X'_{t+1}) \\ &\quad - \big(\mathcal{H}(t, X_t, Y_{t+1}, Z_t, \mu_t, \psi_t) - \mathcal{H}(t, X'_t, Y'_{t+1}, Z'_t, \mu'_t, \psi'_t)\big) \\ &\quad - \sigma^\top (Z_t - Z'_t) - \big(R(t, X_t, \mu_t, \psi_t) - R(t, X'_t, \mu'_t, \psi'_t)\big) \Big] \\ &= \sum_{t=0}^{T-1} \Big[(Y_{t+1} - Y'_{t+1}) \cdot (X_{t+1} - X'_t) - \big(\mathcal{H}(t, X_t, Y_{t+1}, Z_t, \mu_t, \psi_t) - \mathcal{H}(t, X'_t, Y'_{t+1}, Z'_t, \mu'_t, \psi'_t)\big) \\ &\quad - \sigma^\top (Z_t - Z'_t) - \big(R(t, X_t, \mu_t, \psi_t) - R(t, X'_t, \mu'_t, \psi'_t)\big) \Big] \\ =& \sum_{i=1}^{N} \sum_{t=0}^{T-1} \left[Y_{i,t+1} \cdot X_{i,t+1} - Y'_{i,t+1} \cdot X'_{i,t+1} \right] \\ &- \sum_{i=1}^{N} \sum_{t=0}^{T-1} \left[\mathcal{H}_i(t, X_{i,t}, Y_{i,t+1}, \mu_t, \psi_{i,t}) - \mathcal{H}_i(t, X'_{i,t}, Y'_{i,t+1}, \mu'_t, \psi'_{i,t}) \right] \\ &- \frac{1}{N} \sum_{i=1}^{N} \sum_{t=0}^{T-1} \left[R_i(t, X_{i,t}, \mu_t, \psi_{i,t}) - R_i(t, X'_{i,t}, \mu'_t, \psi'_{i,t}) \right]. \end{aligned} \tag{44}$$

where we omit the $\sigma^\top (Z_t - Z'_t)$ term since $Z_t = Z'_t$, reflecting that both paths are driven by the same Wiener process.

The first term on the right-hand side can be rewritten as

$$\begin{aligned} &\sum_{i=1}^{N} \sum_{t=0}^{T-1} \left[Y_{i,t+1} \cdot X_{i,t+1} - Y'_{i,t+1} \cdot X'_{i,t+1} \right] \\ =& \sum_{i=1}^{N} \sum_{t=0}^{T-1} \left[Y_{i,t+1} \cdot \Delta X_{i,t+1} + X_{i,t+1} \cdot \Delta Y_{i,t+1} - \Delta X_{i,t+1} \cdot \Delta Y_{i,t+1} \right], \end{aligned} \tag{45}$$

where we have defined $\Delta X_{i,t} := X_{i,t} - X'_{i,t}$ and $\Delta Y_{i,t} := Y_{i,t} - Y'_{i,t}$. Further simplification is possible by noting that $\Delta X_{i,0} = 0$, and thus

$$\begin{aligned}
&\sum_{i=1}^{N}\sum_{t=0}^{T-1}\left[Y_{i,t+1}\cdot\Delta X_{i,t+1} + X_{i,t+1}\cdot\Delta Y_{i,t+1}\right]\\
=&\sum_{i=1}^{N}\left\{Y_{i,T}\cdot\Delta X_{i,T} + \sum_{t=0}^{T-1}\left[Y_{i,t}\cdot\Delta X_{i,t} + X_{i,t+1}\cdot\Delta Y_{i,t+1}\right]\right\}\\
=&\sum_{i=1}^{N}\left\{Y_{i,T}\cdot\Delta X_{i,T} + \sum_{t=0}^{T-1}\left[\nabla_x\mathcal{H}_i(t,X_{i,t},Y_{i,t+1},\mu_t,\psi'_{i,t})\cdot\Delta X_{i,t} + \nabla_y\mathcal{H}_i(t,X_{i,t},Y_{i,t+1},\mu_t,\psi'_{i,t})\cdot\Delta Y_{i,t+1}\right]\right\}
\end{aligned}$$

Introducing $P_{i,t} := (X_{i,t}, Y_{i,t+1})$ and $P'_{i,t} := (X'_{i,t}, Y'_{i,t+1})$, this expression can be reformulated as

$$\sum_{i=1}^{N}\sum_{t=0}^{T-1}\left[Y_{i,t+1}\cdot\Delta X_{i,t+1} + X_{i,t+1}\cdot\Delta Y_{i,t+1}\right] = \sum_{i=1}^{N}\left\{Y_{i,T}\cdot\Delta X_{i,T} + \sum_{t=0}^{T-1}\nabla_P\mathcal{H}_i(t,P_{i,t},\mu_t,\psi_{i,t})\cdot\Delta P_{i,t}\right\} \tag{46}$$

Similarly, we also have

$$\begin{aligned}
&\sum_{i=1}^{N}\sum_{t=0}^{T-1}\Delta X_{i,t+1}\cdot\Delta Y_{i,t+1}\\
=&\frac{1}{2}\sum_{i=1}^{N}\sum_{t=0}^{T-1}\left[\Delta X_{i,t+1}\cdot\Delta Y_{i,t+1} + \Delta X_{i,t+1}\cdot\Delta Y_{i,t+1}\right]\\
=&\frac{1}{2}\sum_{i=1}^{N}\left\{\Delta X_{i,T}\cdot\Delta Y_{i,T} + \frac{1}{2}\sum_{t=0}^{T-1}\left[\nabla_p\mathcal{H}_i(t,P_{i,t},\mu_t,\psi_{i,t}) - \nabla_p\mathcal{H}_i(t,P'_{i,t},\mu_t,\psi'_{i,t})\right]\cdot\Delta P_{i,t}\right\}\\
=&\frac{1}{2}\sum_{i=1}^{N}\left\{\Delta X_{i,T}\cdot\Delta Y_{i,T} + \frac{1}{2}\sum_{t=0}^{T-1}\left[\nabla_p\mathcal{H}(t,P_{i,t},Z_t,\mu_t,\psi_t) - \nabla_z\mathcal{H}(t,P_{i,t},Z_t,\mu_t,\psi'_{i,t})\right]\cdot\Delta P_{i,t}\right.\\
&\left.+\ \frac{1}{2}\sum_{t=0}^{T-1}\Delta P_{i,t}\cdot\nabla_z^2\mathcal{H}(t,P_{i,t}+r_1(t)\Delta P_{i,t},Z_t,\mu_t,\psi_{i,t})\Delta P_{i,t}\right\}
\end{aligned} \tag{47}$$

where in the last line, Taylor's theorem was applied with $r_1(t) \in [0,1]$ for each t. Consequently, the terminal terms in Eq.(46) and Eq.(47) can be rewritten as follows

$$\begin{aligned}
&(Y'_{i,T} + \frac{1}{2}\Delta Y_{i,T})\cdot\Delta X_{i,T}\\
=&-\frac{1}{N}\nabla\mathcal{V}(\Phi(\cdot);\ \cdot)\Phi(\cdot,X'_{i,T})\cdot\Delta X_{i,T} - \frac{1}{2N}\left[\nabla\mathcal{V}(\Phi(\cdot);\ \cdot)\Phi(\cdot,X_{i,T}) - \nabla\mathcal{V}(\Phi(\cdot);\ \cdot)\Phi(\cdot,X'_{i,T})\right]\cdot\Delta X_{i,T}\\
=&-\frac{1}{N}\nabla\mathcal{V}(\Phi(\cdot);\ \cdot)\Phi(\cdot,X'_{i,T})\cdot\Delta X_{i,T} - \frac{1}{2N}\Delta X_{i,T}\cdot\nabla^2\mathcal{V}(\Phi(\cdot);\ \cdot)\Phi(\cdot,X'_{i,T}+r_2\Delta X_{i,T})\Delta X_{i,T}\\
=&-\frac{1}{N}(\Phi(\cdot,X_T) - \Phi(\cdot,X'_T)) - \frac{1}{2N}\Delta X_{i,T}\cdot[\nabla^2\Phi(\cdot,X'_{i,T}+r_2\Delta X_{i,T}) + \nabla^2\Phi(\cdot,X'_{i,T}+r_3\Delta X_{i,T})]\Delta X_{i,T},
\end{aligned} \tag{48}$$

for some $r_2, r_3 \in [0,1]$. Finally, for each $t = 0, 1, \ldots, T-1$, we have

$$\begin{aligned}
&\mathcal{H}(t,P_{i,t},Z_{i,t},\mu_t,\psi_{i,t}) - \mathcal{H}(t,P'_{i,t},Z'_{i,t},\mu'_t\psi'_{i,t})\\
=&\left[\mathcal{H}(t,P'_{i,t},Z'_{i,t},\mu'_t,\psi_{i,t}) - \mathcal{H}(t,P_{i,t},Z_{i,t},\mu_t,\psi'_{i,t})\right]\\
&+\nabla_p\mathcal{H}(t,P'_{i,t},Z_{i,t},\mu_t,\psi_{i,t})\cdot\Delta P_{i,t}\\
&+\frac{1}{2}\Delta P_{i,t}\cdot\nabla_p^2\mathcal{H}(t,P'_{i,t}+r_4(t)\Delta P_{i,t},Z_{i,t},\mu_t,\psi_{i,t})\Delta P_{i,t}
\end{aligned} \tag{49}$$

where $r_4(t) \in [0,1]$.

Substituting Eq.(45)-(49) into Eq.(44) yields

$$\begin{aligned}
&\frac{1}{N}\sum_{i=1}^{N}\left[\sum_{t=0}^{T-1}R_i(t,X_{i,t},\mu_t,\psi_{i,t}) + \mathcal{V}(\Phi_i(\cdot);\ \cdot)\Phi_i(X_{i,T},\psi_{i,T})\right]\\
&-\frac{1}{N}\sum_{i=1}^{N}\left[\sum_{t=0}^{T-1}R_i(t,X'_{i,t},\mu'_t,\psi'_{i,t}) + \mathcal{V}(\Phi_i(\cdot);\ \cdot)\Phi_i(X'_{i,T},\psi'_{i,T})\right]
\end{aligned}$$

$$
\begin{aligned}
&= -\sum_{i=1}^{N}\sum_{t=0}^{T-1}\left[\mathcal{H}_i(t, X'_t, Y'_{t+1}, \mu'_t, \psi_t) - \mathcal{H}_i(t, X'_t, Y'_{t+1}, \mu'_t, \psi'_{i,t})\right] \\
&\quad + \frac{1}{2N}\sum_{i=1}^{N}\left\{\Delta X_{i,T}\cdot \mathcal{V}(\Phi_i(\cdot);\ \cdot)\left[\nabla^2\Phi_i(\cdot, X'_{i,T} + r_2\Delta X_{i,T}) + \nabla^2\Phi_i(\cdot, X'_{i,T} + r_3\Delta X_{i,T})\right]\Delta X_{i,T}\right\} \\
&\quad + \frac{1}{2}\sum_{i=1}^{N}\sum_{t=0}^{T-1}\left[\nabla_p\mathcal{H}_i(\cdot, P'_{i,t}, \psi_{i,t}) - \nabla_p\mathcal{H}_i(\cdot, P'_{i,t}, \psi'_{i,t})\right]\cdot \Delta P_{i,t} \\
&\quad + \frac{1}{2}\sum_{i=1}^{N}\sum_{t=0}^{T-1}\Delta P_{i,t}\cdot\left[\nabla_p^2\mathcal{H}_i(\cdot, P'_{i,t} + r_1(t)\Delta P_{i,t}) - \nabla_p^2\mathcal{H}_i(\cdot, P'_{i,t} + r_4(t)\Delta P_{i,t})\right]\Delta P_{i,t}. \qquad (50)
\end{aligned}
$$

Observe that by summing over all s, the left-hand side reduces to $\mathcal{L}(\psi) - \mathcal{L}(\psi')$. Next, we simplify the right-hand side. First, applying Assumption 1, we obtain

$$
\Delta X_{i,T}\cdot \mathcal{V}(\Phi_i(\cdot);\ \cdot)\left[\nabla^2\Phi(\cdot, X'_{i,T} + r_2\Delta X_{i,T}) + \nabla^2\Phi(\cdot, X'_{i,T} + r_3\Delta X_{i,T})\right]\Delta X_{i,T} \le K\|\Delta X_{i,T}\|_2^2. \qquad (51)
$$

Next

$$
\begin{aligned}
&\left[\nabla_z\mathcal{H}(\cdot, P'_{i,t}, \psi_{i,t}) - \nabla_z\mathcal{H}(\cdot, P'_{i,t}, \psi'_{i,t})\right]\cdot\Delta P_{i,t} \\
\le&\|\nabla_x\mathcal{H}(\cdot, X'_{i,t}, Y'_{i,t+1}, \psi_{i,t}) - \nabla_x\mathcal{H}(\cdot, X'_{i,t}, Y'_{i,t+1}, \psi'_{i,t})\|\|\Delta X_{i,t}\| \\
&+ \|\nabla_y\mathcal{H}(\cdot, X'_{i,t}, Y'_{i,t+1}, \psi_{i,t}) - \nabla_y\mathcal{H}(\cdot, X'_{i,t}, Y'_{i,t+1}, \psi'_{i,t})\|\|\Delta Y_{i,t+1}\| \\
\le&\frac{1}{2N}\|\Delta X_{i,t}\|^2 + \frac{N}{2}\|\nabla_x\mathcal{H}(\cdot, X'_{i,t}, Y'_{i,t+1}, \psi_t) - \nabla_x\mathcal{H}(\cdot, X'_{i,t}, Y'_{i,t+1}, \psi'_{i,t})\|_2^2 \\
&+ \frac{N}{2}\|\Delta Y_{i,t}\|_2^2 + \frac{1}{2N}\|\nabla_y\mathcal{H}(\cdot, X'_{i,t}, Y'_{i,t+1}, \psi_{i,t}) - \nabla_y\mathcal{H}(\cdot, X'_{i,t}, Y'_{i,t+1}, \psi'_{i,t})\|_2^2 \\
\le&\frac{1}{2N}\|\Delta X_{i,t}\|_2^2 + \frac{C^2}{2N}\|\nabla_x b(\cdot, X'_{i,t}, \psi_{i,t}) - \nabla_x b(\cdot, X'_{i,t}, \psi'_{i,t})\|_2^2 \\
&+ \frac{1}{2N}\|\nabla_x R(\cdot, X'_{i,t}, \psi_{i,t}) - \nabla_x R(\cdot, X'_{i,t}, \psi'_{i,t})\|_2^2 \\
&+ \frac{N}{2}\|\Delta Y_{i,t}\|_2^2 + \frac{1}{2N}\|b(\cdot, X'_{i,t}, \psi_{i,t}) - b(\cdot, X'_{i,t}, \psi'_{i,t})\|_2^2, \qquad (52)
\end{aligned}
$$

where in the last line, we utilized Lemma 5. Similarly, the last term in Eq.(50) can be simplified. Since the second derivative of $\mathcal{H}$ with respect to Y vanishes due to its linearity, following the same reasoning as in Eq.(51) and applying Lemma 5, we obtain

$$
\begin{aligned}
&\Delta P_{i,t}\cdot\left[\nabla_z^2\mathcal{H}(\cdot, P'_{i,t} + r_1(t)\Delta P_{i,t}) - \nabla_z^2\mathcal{H}(\cdot, P'_{i,t} + r_4(t)\Delta P_{i,t})\right]\Delta P_{i,t} \\
\le&\frac{2KC}{N}\|\Delta X_{i,t}\|^2 + 4K\|\Delta X_{i,t}\|\|\Delta Y_{i,t+1}\| \\
\le&\frac{2KC}{N}\|\Delta X_{i,t}\|^2 + \frac{2K}{N}\|\Delta X_{i,t}\|^2 + 2KN\|\Delta Y_{i,t+1}\|^2 \qquad (53)
\end{aligned}
$$

Substituting Eq.(51)-(53) into Eq.(50), as follow

$$
\begin{aligned}
&\frac{1}{N}\sum_{i=1}^{N}\left[\sum_{t=0}^{T-1} R_i(t, X_{i,t}, \mu_t, \psi_{i,t}) + \mathcal{V}(\Phi_i(\cdot);\ \theta)\Phi_i(t, X_{i,T}, \mu_t, \psi_{i,T})\right] \\
&- \frac{1}{N}\sum_{i=1}^{N}\left[\sum_{t=0}^{T-1} R_i(t, X'_{i,t}, \mu'_t, \psi'_{i,t}) + \mathcal{V}(\Phi_i(\cdot);\ \theta)\Phi_i(t, X'_{i,T}, \mu'_t, \psi'_{i,T})\right] \\
=& -\sum_{i=1}^{N}\sum_{t=0}^{T-1}\left[\mathcal{H}_i(t, X'_t, Y'_{t+1}, \mu'_t, \psi_t) - \mathcal{H}_i(t, X'_t, Y'_{t+1}, \mu'_t, \psi'_{i,t})\right] \\
&+ \frac{C}{N}\sum_{i=1}^{N}\sum_{t=0}^{T}\|\Delta X_{i,t}\|^2 + CN\sum_{i=1}^{N}\sum_{t=0}^{T-1}\|\Delta Y_{i,t+1}\|^2 \\
&+ \frac{C}{N}\sum_{i=1}^{N}\sum_{t=0}^{T-1}\|b_i(t, X'_{i,t}, \mu'_t, \psi_{i,t}) - b_i(t, X'_{i,t}, \mu'_t, \psi'_{i,t})\|^2 \\
&+ \frac{C}{N}\sum_{i=1}^{N}\sum_{t=0}^{T-1}\|\nabla_x b_i(t, X'_{i,t}, \mu'_t, \psi_{i,t}) - \nabla_x b_i(t, X'_{i,t}, \mu'_t, \psi'_{i,t})\|^2
\end{aligned}
$$

$$+\frac{C}{N}\sum_{i=1}^{N}\sum_{t=0}^{T-1}\|\nabla_x R_i(t,X'_{i,t},\mu'_t,\psi_{i,t})-\nabla_x R_i(t,X'_{i,t},\mu'_t,\psi'_{i,t})\|^2 \tag{54}$$

We now proceed to estimate the magnitudes of $\Delta X_{i,t}$ and $\Delta Y_{i,t}$. Notably, since $\Delta X_{i,0}=0$, it follows that for each $t=0,\dots,T-1$, we have

$$\begin{aligned}\|\Delta X_{i,t+1}\| \leq & \|b(t,X_{i,t},\mu_t,\psi_{i,t})-b(t,X'_{i,t},\mu'_t,\psi_{i,t})\|+\|b(t,X'_{i,t},\mu'_t,\psi_{i,t})-b(X'_{i,t},\mu'_t,\psi'_{i,t})\|\\ \leq & K\|\Delta X_{i,t}\|+\|b(t,X'_{i,t},\mu'_t,\psi_{i,t})-b(t,X'_{i,t},\mu'_t,\psi'_{i,t})\|\end{aligned}$$

Using Lemma 4, we have

$$\|\Delta X_{i,t}\|\leq C\sum_{i=1}^{N}\sum_{t=0}^{T-1}\|b(t,X'_{i,t},\mu'_t,\psi_{i,t})-b(t,X'_{i,t},\mu'_t,\psi'_{i,t})\| \tag{55}$$

Similarly

$$\begin{aligned}\|\Delta Y_{i,t}\| \leq & \|\nabla_x\mathcal{H}_i(t,X_{i,t},Y_{i,t+1},Z_{i,t},\mu_t,\psi_{i,t})-\nabla_x\mathcal{H}_i(t,X'_{i,t},Y'_{i,t+1},Z'_{i,t},\mu'_t,\psi'_{i,t})\|\\ \leq & 2K\|\Delta Y_{i,t+1}\|+\frac{C}{N}\|\Delta X_{i,t}\|\\ & +\frac{C}{N}\|\nabla_x b(t,X'_{i,t},\mu'_t,\psi_{i,t})-\nabla_x b(t,X'_{i,t},\mu'_t,\psi'_{i,t})\|\\ & +\frac{C}{N}\|\nabla_x R(t,X'_{i,t},\mu'_t,\psi_{i,t})-\nabla_x R(t,X'_{i,t},\mu'_t,\psi'_{i,t})\|,\end{aligned}$$

and so by Lemma 4, Eq.(55) and the fact that $\|\Delta Y_{i,T}\|\leq\frac{K}{N}\|\Delta X_{i,T}\|$ by Assumption 1, we have

$$\begin{aligned}\|\Delta Y_{i,t}\| \leq & \frac{C}{N}\sum_{i=1}^{N}\sum_{t=0}^{T-1}\|b(t,X'_{i,t},\mu'_t,\psi_{i,t})-b(t,X'_{i,t},\mu'_t,\psi'_{i,t})\|\\ & +\frac{C}{N}\sum_{i=1}^{N}\sum_{t=0}^{T-1}\|\nabla_x b(t,X'_{i,t},\mu'_t,\psi_{i,t})-\nabla_x b(t,X'_{i,t},\mu'_t,\psi'_{i,t})\|\\ & +\frac{C}{N}\sum_{i=1}^{N}\sum_{t=0}^{T-1}\|\nabla_x R(t,X'_{i,t},\mu'_t,\psi_{i,t})-\nabla_x R(t,X'_{i,t},\mu'_t,\psi'_{i,t})\|.\end{aligned} \tag{56}$$

To finalize the proof of Theorem 3, we incorporate the estimates from Eq.(55) and Eq.(56) into Eq.(54). □

8 Proof of NDDV

8.1 Proof of the data state utility function

In this section, we provide detailed proof of the data state utility function based on the dynamic interaction between the data state and the data co-state. Specifically, we address the optimality and effectiveness of the data state utility function.

Theorem 3 Assume that a subset S undergoes a single training session using NDDV, yielding an optimal control strategy ψ^*, and simultaneously obtaining the data state utility functions $U(S)=(U_1(S),U_2(S),\cdots,U_n(S))$ corresponding to all data points. The data state utility $U(S)$ for the data point (x_i,y_i) is then as follows

$$U_i(S)=-X_{i,T}\cdot Y_{i,T},$$

Proof For optimality, we set the control strategy ψ during optimization and the optimal control strategy ψ^*. Compared to ψ, ψ^* possesses, in some sense, the optimal descent direction to obtain the best data state utility functions $U(S)$.

According to the Hamiltonian maximization condition, as

$$\mathcal{H}(t,X^*_t,Y^*_t,Z^*_t,\psi^*_t)-\mathcal{H}(t,X^*_t,Y^*_t,Z^*_t,\psi)\geq 0, \tag{57}$$

where $\mathcal{H}$ is

$$\mathcal{H}(t,X_t,Y_t,Z_t,\psi_t)=b(t,X_t,\psi_t)\cdot Y_t+\mathrm{tr}(\sigma^\top Z_t)-R(t,X_t,\psi_t), \tag{58}$$

The rate of change for the Hamiltonian is given

$$\frac{\partial \mathcal{H}}{\partial \psi_t} = \frac{\partial b(t, X_t, \psi_t)}{\partial \psi_t} \cdot Y_t - \frac{\partial R(t, X_t, \psi_t)}{\partial \psi_t}. \tag{59}$$

For arbitrary ψ, we have

$$b(t, X_t^*, \psi_t^*) \cdot Y_t^* - R(t, X_t^*, \psi_t^*) \geq b(t, X_t^*, \psi_t) \cdot Y_t^* - R(t, X_t^*, \psi_t). \tag{60}$$

Consequently, X_t and Y_t evolve along the optimal trajectories X_t^* and Y_t^*, leading to $U(S, \psi^*) = -X_T^* \cdot Y_T^*$ that is superior to $U(S, \psi)$ obtained under arbitrary ψ. For an arbitrary data point (x_i, y_i), $U_i(S, \psi^*)$ is superior to $U_i(S, \psi)$.

To compare the utility functions $U_i(S, \psi^*)$ and $U_i(S, \psi)$ for (x_i, y_i), we first consider the difference of these

$$\begin{aligned} \Delta U_i(S) &= U_i(S, \psi^*) - U_i(S, \psi) \\ &= -X_{i,T}^* \cdot Y_{i,T}^* + X_{i,T} \cdot Y_{i,T}, \end{aligned} \tag{61}$$

Substituting $X_{i,T}^* = X_{i,T} + \Delta X_{i,T}$ and $Y_{i,T}^* = Y_{i,T} + \Delta Y_{i,T}$, we get

$$\Delta U_i(S) = -(X_{i,T} + \Delta X_{i,T}) \cdot (Y_{i,T} + \Delta Y_{i,T}) + X_{i,T} \cdot Y_{i,T}, \tag{62}$$

Expanding and simplifying

$$\begin{aligned} \Delta U_i(S) &= -X_{i,T} \cdot Y_{i,T} - X_{i,T} \cdot \Delta Y_{i,T} - \Delta X_{i,T} \cdot Y_{i,T} - \Delta X_{i,T} \cdot \Delta Y_{i,T} + X_{i,T} \cdot Y_{i,T} \\ &= -X_{i,T} \cdot \Delta Y_{i,T} - \Delta X_{i,T} \cdot Y_{i,T} - \Delta X_{i,T} \cdot \Delta Y_{i,T}, \end{aligned} \tag{63}$$

Since ψ^* maximizes the Hamiltonian, it satisfies

$$\frac{\partial \mathcal{H}}{\partial \psi_t}\Big|_{\psi_t = \psi_t^*} = \frac{\partial b(t, X_t^*, \psi_t^*)}{\partial \psi_t} \cdot Y_t^* - \frac{\partial R(t, X_t^*, \psi_t^*)}{\partial \psi_t} = 0, \tag{64}$$

From Eq.(64), the ψ^* satisfies

$$\frac{\partial \mathcal{H}}{\partial \psi_t}\Big|_{\psi_t = \psi_t^*} \geq \frac{\partial \mathcal{H}}{\partial \psi_t}\Big|_{\psi_t = \psi_t}, \tag{65}$$

which implies

$$\frac{\partial b(t, X_t^*, \psi_t^*)}{\partial \psi_t} \cdot Y_t^* - \frac{\partial R(t, X_t^*, \psi_t^*)}{\partial \psi_t} \geq \frac{\partial b(t, X_t^*, \psi_t)}{\partial \psi_t} \cdot Y_t^* - \frac{\partial R(t, X_t^*, \psi_t)}{\partial \psi_t}. \tag{66}$$

Furthermore, we consider the rates of change for $X_{i,t}$ and $Y_{i,t}$, as follows

$$\begin{cases} \mathrm{d}\Delta X_{i,t} = b(t, X_{i,t}^*, \psi_{i,t}^*)\mathrm{d}t - b(t, X_{i,t}, \psi_t)\mathrm{d}t, \\ \mathrm{d}\Delta Y_{i,t} = -\left[\nabla_x \mathcal{H}(t, X_{i,t}^*, Y_{i,t}^*, \psi_{i,t}^*) - \nabla_x \mathcal{H}(t, X_{i,t}, Y_{i,t}, \psi_{i,t})\right] \mathrm{d}t + (Z_{i,t}^* - Z_{i,t})\mathrm{d}W_t, \end{cases} \tag{67}$$

Similarly, the changes in $\Delta X_{i,T}$ and $\Delta Y_{i,T}$ at the terminal time T are given

$$\begin{cases} \Delta X_{i,T} = \Delta X_{i,t} + \int_t^T \left[b(s, X_{i,s}^*, \psi_{i,s}^*) - b(s, X_{i,s}, \psi_{i,s})\right] \mathrm{d}s, \\ \Delta Y_{i,T} = \Delta Y_{i,t} - \int_t^T \left[\nabla_x \mathcal{H}(s, X_{i,s}^*, Y_{i,s}^*, \psi_{i,s}^*) + \nabla_x \mathcal{H}(s, X_{i,s}, Y_{i,s}, \psi_{i,s})\right] \mathrm{d}s + \int_t^T (Z_{i,s}^* - Z_{i,s})\mathrm{d}W_s. \end{cases} \tag{68}$$

We then compute $\Delta X_{i,T} \cdot \Delta Y_{i,T}$ by expanding the following terms

$$\begin{aligned} &\Delta X_{i,T} \cdot \Delta Y_{i,T} \\ =&\Delta X_{i,t} \cdot \Delta Y_{i,t} + \Delta X_{i,t} \cdot \int_t^T (Z_{i,s}^* - Z_{i,s})\mathrm{d}W_s \\ &- \Delta X_{i,t} \cdot \int_t^T \left(\nabla_x \mathcal{H}(s, X_{i,s}^*, Y_{i,s}^*, \psi_{i,s}^*) + \nabla_x \mathcal{H}(s, X_{i,s}, Y_{i,s}, \psi_{i,s})\right) \mathrm{d}s \\ &+ \left(\int_t^T \left[b(s, X_{i,s}^*, \psi_{i,s}^*) - b(s, X_{i,s}, \psi_{i,s})\right] \mathrm{d}s\right) \cdot \Delta Y_{i,t} \\ &- \left(\int_t^T \left[b(s, X_{i,s}^*, \psi_{i,s}^*) + b(s, X_{i,s}, \psi_{i,s})\right] \mathrm{d}s\right) \cdot \int_t^T \nabla_x \mathcal{H}(s, X_{i,s}^*, Y_{i,s}^*, \psi_{i,s}^*) + \nabla_x \mathcal{H}(s, X_{i,s}, Y_{i,s}, \psi_{i,s})\mathrm{d}s \\ &+ \left(\int_t^T \left[b(s, X_{i,s}^*, \psi_{i,s}^*) - b(s, X_{i,s}, \psi_{i,s})\right] \mathrm{d}s\right) \cdot \int_t^T (Z_{i,s}^* - Z_{i,s})\mathrm{d}W_s, \end{aligned} \tag{69}$$

Using Itô's Isometry to handle the stochastic integral term and applying the properties of $\mathcal{H}$, we simplify

$$\begin{aligned}&\Delta X_{i,T}\cdot\Delta Y_{i,T}\\=&\Delta X_{i,t}\cdot\Delta Y_{i,t}\\&+\int_t^T\left[b(s,X^*_{i,s},\psi^*_{i,s})-b(s,X_{i,s},\psi_{i,s})\right]\\&\quad\cdot\left[-\nabla_x\mathcal{H}(s,X^*_{i,s},Y^*_{i,s},\psi^*_{i,s})+\nabla_x\mathcal{H}(s,X_{i,s},Y_{i,s},\psi_{i,s})\right]\mathrm{d}s.\end{aligned}\tag{70}$$

According to ψ^*, $\mathcal{H}$ is maximized:

$$b(s,X^*_{i,s},\psi^*_{i,s})\cdot\left[-\nabla_x\mathcal{H}(s,X^*_{i,s},Y^*_{i,s},\psi^*_{i,s})\right]\geq b(s,X_{i,s},\psi_{i,s})\cdot\left[-\nabla_x\mathcal{H}(s,X_{i,s},Y_{i,s},\psi_{i,s})\right].\tag{71}$$

Thus

$$\Delta X_{i,T}\cdot\Delta Y_{i,T}\leq\Delta X_{i,t}\cdot\Delta Y_{i,t}.\tag{72}$$

It can be observed that, in the process of ψ searching for the optimal control strategy, the data state and co-state gradually stabilize.

Integrating from t to T yields

$$\Delta X_{i,T}\cdot\Delta Y_{i,T}\leq 0.\tag{73}$$

Therefore, Eq.(63) satisfies

$$\Delta U_i(S)\geq -X_{i,T}\cdot\Delta Y_{i,T}-\Delta X_{i,T}\cdot Y_{i,T}\geq 0.\tag{74}$$

Thus

$$U_i(S,\psi^*)\geq U_i(S,\psi).\tag{75}$$

When $\psi\neq\psi^*$, we typically have

$$-X_{i,T}\cdot\Delta Y_{i,T}-\Delta X_{i,T}\cdot Y_{i,T}>0,\tag{76}$$

which implies the strict inequality

$$U_i(S,\psi^*){>}U_i(S,\psi).\tag{77}$$

The proof of optimality is complete.

For effectiveness, for arbitrary two arbitrary data points (x_i,y_i) and (x_j,y_j) in the subset S, where $i\neq j\in[N]$, if $\mathcal{V}(\Phi_i(\cdot);\ \theta){>}\mathcal{V}(\Phi_j(\cdot);\ \theta)$, we compare $U_i(S)$ and $U_j(S)$.

Given that $\mathcal{V}(\Phi_i(\cdot);\ \theta){>}\mathcal{V}(\Phi_j(\cdot);\ \theta)$, it follows that

$$\begin{cases}Y_{i,T}=-\nabla_x\left(\mathcal{V}(\Phi_i(\cdot);\ \theta)\Phi_i(X_{i,T},\mu_T,\psi_{i,T})\right),\\Y_{j,T}=-\nabla_x\left(\mathcal{V}(\Phi_j(\cdot);\ \theta)\Phi_j(X_{j,T},\mu_T,\psi_{j,T})\right),\end{cases}\tag{78}$$

We can then express the difference in utility functions as

$$U_i(S)-U_j(S)=-X_{i,T}\cdot Y_{i,T}+X_{j,T}\cdot Y_{j,T},\tag{79}$$

Substituting the expressions for $Y_{i,T}$ and $Y_{j,T}$ in Eq.(79), we obtain

$$U_i(S)-U_j(S)=X_{i,T}\cdot\nabla_x(\mathcal{V}(\Phi_i(\cdot);\ \theta)\Phi_i(X_{i,T},\mu_T,\psi_{i,T}))-\tag{80}$$

$$X_{j,T}\cdot\nabla_x(\mathcal{V}(\Phi_j(\cdot);\ \theta)\Phi_j(X_{j,T},\mu_T,\psi_{j,T})),\tag{81}$$

Since $\mathcal{V}(\Phi_i(\cdot);\ \theta){>}\mathcal{V}(\Phi_j(\cdot);\ \theta)$, the data point (x_i,y_i) contributes more to the training than (x_j,y_j). It is well known that training involves minimizing the loss function, and in optimal control, the terminal loss function is major. Thus, $\Phi_i(\cdot){>}\Phi_j(\cdot)$, and the data point (x_i,y_i) is more readily optimized. Based on this, we get

$$\nabla_x(\mathcal{V}(\Phi_i(\cdot);\ \theta)\Phi_i(X_{i,T},\mu_T,\psi_{i,T})){>}\nabla_x(\mathcal{V}(\Phi_j(\cdot);\ \theta)\Phi_j(X_{j,T},\mu_T,\psi_{j,T})),\tag{82}$$

Furthermore, the data point (x_i,y_i) has been better optimized compared to (x_j,y_j), making it easier to achieve a stable terminal data state and data co-state. From Eq.(72), we get

$$\Delta X_{i,T}\cdot\Delta Y_{i,T}{<}\Delta X_{j,T}\cdot\Delta Y_{j,T},\tag{83}$$

Substituting Eq.(83) into Eq.(83), we obtain

$$\Delta X_{i,T}\cdot\Delta\nabla_x(\mathcal{V}(\Phi_i(\cdot);\ \theta)\Phi_i(X_{i,T},\mu_T,\psi_{i,T})){>}\Delta X_{j,T}\cdot\Delta\nabla_x(\mathcal{V}(\Phi_j(\cdot);\ \theta)\Phi_j(X_{j,T},\mu_T,\psi_{j,T})),\tag{84}$$

Based on Eq.(83), we simplify this

$$\Delta X_{i,T}{>}\Delta X_{j,T},\tag{85}$$

Thus, we obtain

$$U_i(S){>}U_j(S).\tag{86}$$

Traditional utility functions $U(S\cup\{(x_i,y_i)\})$ often employ backpropagation-based learning algorithms primarily based on empirical risk minimization to measure model performance in classification or regression tasks. The data state utility function $U_i(S)$ is similar to a performance evaluation mode based on Gradient × Input [37, 38], which has been shown to slightly outperform commonly used backpropagation-based learning algorithms. Based on this, the data state utility function $U_i(S)$ can be viewed as an enhancement of the traditional utility function $U(S\cup\{(x_i,y_i)\})$, such that $\|U_i(S)-U(S\cup\{(x_i,y_i)\})\|\leq\varepsilon$.

The proof of effectiveness is complete. □

8.2 Proof of dynamic marginal contribution

In this section, we provide a detailed proof of dynamic marginal contribution in Proposition 1.

Proposition 1 For arbitrary two data points (x_i, y_i) and (x_j, y_j), $i \neq j \in [N]$, if $U_i(S) > U_j(S)$, then $\Delta(x_i, y_i;\ U_i(S)) > \Delta(x_j, y_j;\ U_j(S))$.

Proof For $i \neq j$, if $U_i(S) > U_j(S)$, we have

$$\begin{aligned}
&\Delta(x_i, y_i;\ U_i(S)) - \Delta(x_j, y_j;\ U_j(S)) \\
&= \left[U_i(S) - \sum_{i' \in \{1,\ldots,N\}\backslash i} \frac{U_{i'}(S)}{N-1}\right] - \left[U_j(S) - \sum_{j' \in \{1,\ldots,N\}\backslash j} \frac{U_{j'}(S)}{N-1}\right] \\
&= \left[U_i(S) - U_j(S)\right] + \frac{1}{N-1}\left[\sum_{j' \in \{1,\ldots,N\}\backslash j} U_{j'}(S) - \sum_{i' \in \{1,\ldots,N\}\backslash i} U_{i'}(S)\right] \\
&= \left[U_i(S) - U_j(S)\right] + \frac{1}{N-1}\left[U_i(S) - U_j(S)\right] \\
&= \frac{N}{N-1}\left[U_i(S) - U_j(S)\right] \\
&> 0.
\end{aligned} \tag{87}$$

Then, the proof is complete. □

8.3 Proof of dynamic data valuation

In this section, we provide a detailed proof process demonstrating how our proposed dynamic data valuation metric satisfies the common axioms.

For the **efficiency** property, If there are no data points within the coalition, then no data states are presented. In this case, it still holds that

$$\begin{aligned}
\sum_{i \in N} \phi(x_i, y_i;\ U_i) &= \sum_{i \in N} \Delta(x_i, y_i;\ U_i) \\
&= \sum_{i \in N}\left[U_i(S) - \sum_{j \in \{1,\ldots,N\}\backslash i} \frac{U_j(S)}{N-1}\right] \\
&= -X_T \cdot Y_T - 0 \\
&= U(N) - U(\emptyset),
\end{aligned} \tag{88}$$

For the **symmetry** property, the value to data point (x_i', y_i') is $\phi((x_i', y_i');\ U_i'(S))$. For arbitrary $S \in N \setminus \{(x_i, y_i), (x_i', y_i')\}$, if $U(S \cup (x_i, y_i)) = U(S \cup (x_i', y_i'))$, the proof of this property as follow

$$\begin{aligned}
\phi(x_i', y_i';\ U_i'(S)) &= \Delta(x_i', y_i';\ U_i'(S)) \\
&= U_i'(S) - \sum_{j \in \{1,\ldots,N\}\backslash\{i',i\}} \frac{U_j(S)}{N-1} \\
&= U_i(S) - \sum_{j \in \{1,\ldots,N\}\backslash\{i,i'\}} \frac{U_j(S)}{N-1} \\
&= \phi(x_i, y_i;\ U_i(S)),
\end{aligned} \tag{89}$$

For the **dummy** property, since $U\left(S \cup (x_i, y_i)\right) = U(S)$ for all $S \in N \setminus (x_i, y_i)$ always holds, the value to the data point (x_i, y_i) is

$$\phi(x_i, y_i;\ U_i(S)) = \Delta(x_i, y_i;\ U_i(S))$$

$$
\begin{aligned}
&= U_i(S) - \sum_{j\in\{1,\dots,N\}\backslash i} \frac{U_j(S)}{N-1} \\
&= 0,
\end{aligned}
\tag{90}
$$

For the **additivity** property, we choose the two utility functions $U_1(S)$ and $U_2(S)$ together, impacting the data point (x_i, y_i). The value to it for arbitrary $\alpha_1, \alpha_2 \in \mathbb{R}$ is

$$
\begin{aligned}
&\phi\left(x_i, y_i;\ \alpha_1 U_{1,i}(S) + \alpha_2 U_{2,i}(S)\right) \\
&= \Delta(x_i, y_i;\ \alpha_1 U_{1,i}(S) + \alpha_2 U_{2,i}(S)) \\
&= [\alpha_1 U_{1,i}(S) + \alpha_2 U_{2,i}(S)] - \sum_{j\in\{1,\dots,N\}\backslash i} \frac{\alpha_1 U_{1,j}(S) + \alpha_2 U_{2,j}(S)}{N-1} \\
&= \alpha_1 \left[U_{1,i}(S) - \sum_{j\in\{1,\dots,N\}\backslash i} \frac{U_{1,j}(S)}{N-1}\right] + \alpha_2 \left[U_{2,i}(S) - \sum_{j\in\{1,\dots,N\}\backslash i} \frac{U_{2,j}(S)}{N-1}\right] \\
&= \alpha_1 \phi\left((x_i, y_i);\ U_{1,i}(S)\right) + \alpha_2 \phi\left((x_i, y_i);\ U_{2,i}(S)\right),
\end{aligned}
\tag{91}
$$

For the **marginalism** property, if each data point has the identical marginal impact in two utility functions U_1, U_2, satisfies $U_1(S \cup (x_i, y_i)) - U_1(S) = U_2(S \cup (x_i, y_i)) - U_2(S)$. The proof of this property is as follows

$$
\begin{aligned}
\phi(x_i, y_i;\ U_{1,i}(S)) &= \Delta(x_i, y_i, U_{1,i}(S)) \\
&= U_{1,i}(S) - \sum_{j\in\{1,\dots,N\}\backslash i} \frac{U_{1,j}(S)}{N-1} \\
&= U_1(S \cup (x_i, y_i)) - U_1(S) \\
&= U_2(S \cup (x_i, y_i)) - U_2(S) \\
&= U_{2,i}(S) - \sum_{j\in\{1,\dots,N\}\backslash i} \frac{U_{2,j}(S)}{N-1} \\
&= \phi(x_i, y_i;\ U_{2,i}).
\end{aligned}
\tag{92}
$$

Then, the proof of the five common axioms is complete.

It is evident that our proposed dynamic data valuation metric satisfies the common axioms. Then, we provide the following Proposition 2 to identify important data points among two arbitrarily different data points.

Proposition 2 For arbitrary two data points (x_i, y_i) and (x_j, y_j), $i \neq j \in [N]$, if $U_i(S) > U_j(S)$, then $\phi(x_i, y_i;\ U_i(S)) > \phi(x_j, y_j;\ U_j(S))$.

Proof For $i \neq j$, if $U_i(S) > U_j(S)$, we have

$$
\begin{aligned}
&\phi(x_i, y_i;\ U_i(S)) - \phi(x_j, y_j;\ U_j(S)) \\
&= \Delta(x_i, y_i;\ U_i(S)) - \Delta(x_j, y_j;\ U_j(S)) \\
&= \left[U_i(S) - \sum_{i'\in\{1,\dots,N\}\backslash i} \frac{U_{i'}(S)}{N-1}\right] - \left[U_j(S) - \sum_{j'\in\{1,\dots,N\}\backslash j} \frac{U_{j'}(S)}{N-1}\right] \\
&= \left[U_i(S) - U_j(S)\right] + \frac{1}{N-1}\left[\sum_{j'\in\{1,\dots,N\}\backslash j} U_{j'}(S) - \sum_{i'\in\{1,\dots,N\}\backslash i} U_{i'}(S)\right] \\
&= \left[U_i(S) - U_j(S)\right] + \frac{1}{N-1}\left[U_i(S) - U_j(S)\right]
\end{aligned}
$$

$$= \frac{N}{N-1}\left[U_i(S) - U_j(S)\right]$$
$$> 0. \tag{93}$$

Then, the proof is complete. □

Moreover, the following Proposition 3 shows that our proposed dynamic data valuation metric converges to the LOO metric.

Proposition 3 For arbitrary $i \in [n]$ and error bound ε, if $\|U_i(S) - U(S \cup (x_i, y_i))\| \le \varepsilon$, then $\|\text{NDDV} - \text{LOO}\| \le 2\varepsilon$.

Proof For arbitrary pair data points (x_i, y_i) and (x_j, y_j), $i \neq j \in [N]$. The LOO metric difference between those data points is

$$\begin{aligned}&\phi_{\text{loo}}(x_i, y_i;\ U(S)) - \phi_{\text{loo}}(x_j, y_j;\ U(S))\\&= [U(S \cup (x_i, y_i)) - U(S)] - \left[U(S \cup (x_j, y_j)) - U(S)\right]\\&= U(S \cup (x_i, y_i)) - U(S \cup (x_j, y_j)),\end{aligned} \tag{94}$$

Then, call for Eq.(87), we have

$$\phi_{\text{nddv}}(x_i, y_i;\ U_i(S)) - \phi_{\text{nddv}}(x_j, y_j;\ U_j(S)) = \frac{N}{N-1}\left[U_i(S) - U_j(S)\right] \tag{95}$$

In addition, the L_2 error bound between the NDDV and LOO metric is

$$\begin{aligned}&\|\text{NDDV} - \text{LOO}\|\\&= \|\left[\phi_{\text{nddv}}((x_i, y_i), U_i(S)) - \phi_{\text{nddv}}((x_j, y_j), U_j(S))\right] - \left[\phi_{\text{loo}}((x_i, y_i), U(S)) - \phi_{\text{loo}}((x_j, y_j), U(S))\right]\|\\&= \|\frac{N}{N-1}\left[U_i(S) - U_j(S)\right] - \left[U(S \cup (x_i, y_i)) - U(S \cup (x_j, y_j))\right]\|\\&= \|\left[U_i(S) - U(S \cup (x_i, y_i))\right] - \left[U_j(S) - U(S \cup (x_j, y_j))\right] + \frac{1}{N-1}\left[U_i(S) - U_j(S)\right]\|\\&\le \|U_i(S) - U(S \cup (x_i, y_i))\| + \|U_j(S) - U(S \cup (x_j, y_j))\| + \frac{1}{N-1}\|U_i(S) - U_j(S)\|\\&= 2\varepsilon, \qquad N \to \infty.\end{aligned} \tag{96}$$

Concludes the proof. □

Finally, we also provide the L_2 error bound between our proposed dynamic data valuation metric and Shapley value metric in the following Proposition 4.

Proposition 4 For arbitrary $i \in [N]$ and error bound ε, if $\|U_i(S) - U(S \cup (x_i, y_i))\| \le \varepsilon$, then $\|\text{NDDV} - \text{Shap}\| \le \frac{2\varepsilon N}{N-1}$.

Proof For arbitrary pair data points (x_i, y_i) and (x_j, y_j), $i \neq j \in [N]$. The Shapley value metric difference between those data points is

$$\begin{aligned}&\phi_{\text{shap}}(x_i, y_i;\ U(S)) - \phi_{\text{shap}}(x_i, y_i;\ U(S))\\&= \frac{1}{N}\sum_{i'=1}^{N} \Delta_{i'}(x_i, y_i;\ U(S)) - \frac{1}{N}\sum_{j'=1}^{N} \Delta_{j'}(x_i, y_i;\ U(S))\\&= \sum_{S \in \mathcal{D}^{\setminus(x_i, y_i)}} \frac{|S|!(N-|S|-1)!}{N!}\left[U(S \cup (x_i, y_i)) - U(S)\right]\\&- \sum_{S \in \mathcal{D}^{\setminus(x_j, y_j)}} \frac{|S|!(N-|S|-1)!}{N!}\left[U(S \cup (x_j, y_j)) - U(S)\right]\\&= \sum_{S \in \mathcal{D}^{\setminus(x_i, y_i)}} \frac{|S|!(N-|S|-1)!}{N!}\left[U(S \cup (x_i, y_i)) - U(S \cup (x_j, y_j))\right]\end{aligned}$$

$$\begin{aligned}
&+\sum_{S\in\{T|T\in\mathcal{D},(x_i,y_i)\notin T,(x_j,y_j)\in T\}}\frac{|S|!(N-|S|-1)!}{N!}\left[U(S\cup(x_i,y_i))-U(S)\right]\\
&-\sum_{S\in\{T|T\in\mathcal{D},(x_i,y_i)\in T,(x_j,y_j)\notin T\}}\frac{|S|!(N-|S|-1)!}{N!}\left[U(S\cup(x_j,y_j))-U(S)\right]\\
=&\sum_{S\in\mathcal{D}\backslash\{(x_i,y_i),(x_j,y_j)\}}\frac{|S|!(N-|S|-1)!}{N!}\left[U(S\cup(x_i,y_i))-U(S\cup(x_j,y_j))\right]\\
&+\sum_{S'\in\mathcal{D}\backslash\{(x_i,y_i),(x_j,y_j)\}}\frac{(\mathsf{S'}+1)!(N-\mathsf{S'}-2)!}{N!}\left[U(S'\cup(x_i,y_i))-U(S'\cup(x_j,y_j))\right]\\
=&\sum_{S\in\mathcal{D}\backslash\{(x_i,y_i),(x_j,y_j)\}}\left[\frac{|S|!(N-|S|-1)!}{N!}+\frac{(|S|+1)!(N-|S|-2)!}{N!}\right]\left[U(S\cup(x_i,y_i))-U(S\cup(x_j,y_j))\right]\\
=&\frac{1}{N-1}\sum_{S\in\mathcal{D}\backslash\{(x_i,y_i),(x_j,y_j)\}}\frac{1}{C_{N-2}^{|S|}}\left[U(S\cup(x_i,y_i))-U(S\cup(x_j,y_j))\right]. \qquad (97)
\end{aligned}$$

Then, call for Eq.(87), we have

$$\phi_{\mathrm{nddv}}(x_i,y_i;\,U_i(S))-\phi_{\mathrm{nddv}}(x_j,y_j;\,U_j(S))=\frac{N}{N-1}\left[U_i(S)-U_j(S)\right] \qquad (98)$$

In addition, the L_2 error bound between NDDV and Shapley value metric is

$$\begin{aligned}
&\|\mathrm{NDDV}-\mathrm{Shap}\|\\
&=\|\left[\phi_{\mathrm{nddv}}(x_i,y_i;\,U_i(S))-\phi_{\mathrm{nddv}}(x_j,y_j;\,U_j(S))\right]-\left[\phi_{\mathrm{shap}}(x_i,y_i;\,U(S))-\phi_{\mathrm{shap}}(x_j,y_j;\,U(S))\right]\|\\
&=\left\|\frac{N}{N-1}\left[U_i(S)-U_j(S)\right]-\frac{1}{N-1}\sum_{S\in\mathcal{D}\backslash\{(x_i,y_i),(x_j,y_j)\}}\frac{1}{C_{N-2}^{|S|}}\left[U(S\cup(x_i,y_i))-U(S\cup(x_j,y_j))\right]\right\|\\
&=\left\|\frac{N}{N-1}\left[U_i(S)-U_j(S)\right]-\frac{1}{N-1}\sum_{k=0}^{N-2}C_k^{N-2}\frac{1}{C_{N-2}^{|S|}}\left[U(S\cup(x_i,y_i))-U(S\cup(x_j,y_j))\right]\right\|\\
&\leq\frac{N}{N-1}\left\|\left[U_i(S)-U_j(S)\right]-\min\left\{\sum_{k=0}^{N-2}C_k^{N-2}\frac{1}{NC_{N-2}^{|S|}}\right\}\left[U(S\cup(x_i,y_i))-U(S\cup(x_j,y_j))\right]\right\|\\
&=\frac{N}{N-1}\left\|\left[U_i(S)-U_j(S)\right]-2^{N-2}\frac{1}{NC_{N-2}^{\frac{N-2}{2}}}\left[U(S\cup(x_i,y_i))-U(S\cup(x_j,y_j))\right]\right\|\\
&\leq\frac{N}{N-1}\|U_i(S)-U(S\cup(x_i,y_i))\|+\|U_j(S)-U(S\cup(x_j,y_j))\|\\
&=\frac{2\varepsilon N}{N-1},\qquad N\to\infty. \qquad (99)
\end{aligned}$$

Concludes the proof. □

9 Experiments details

In this section, we evaluate NDDV through three experiments: elapsed time comparison, corrupted data detection, and data points removal/addition. These experiments are designed to assess the computational efficiency, and accuracy in identifying mislabeled data, and the impact of data values on model training, following the evaluation protocols commonly adopted in previous studies [18, 24, 34]. The experiments demonstrate the superior computational efficiency of NDDV, the effectiveness of NDDV in accurately identifying mislabeled or corrupted data points, and the effectiveness of NDDV in improving model performance by selectively removing detrimental (low-value) data points and adding beneficial (high-value) ones. Our Python-based implementation codes are publicly available at https://github.com/liangzhangyong.

9.1 Experimental Setup

We use six datasets that are publicly available in OpenDataVal [39], many of which were used in existing works [18, 19]. We compare NDDV with the following eight methods: LOO [16], DataShapley [18], BetaShapley [19], DataBanzhaf [20], InfluenceFunction [21], KNNShapley [22], AME [40], and Data-OOB [25]. To make our comparison fair, we use the same number or a greater number of utility functions for existing data valuation methods compared to NDDV.

We apply a standard normalization procedure to each dataset, ensuring that each feature has zero mean and unit standard deviation. Next, we divide the normalized dataset into four distinct subsets: training, validation, test, and meta. The training subset is used to train the models, while the validation subset is employed to evaluate the utility functions for the existing data valuation methods. NDDV utilizes only a smaller subset of the training data and does not require the validation subset, as it operates independently of the utility functions. The test subset is reserved for evaluating the test accuracy during the point removal experiments, providing an unbiased assessment of the model's performance on unseen data. The meta subset is utilized for meta-learning purposes, such as hyperparameter tuning or model selection. We fix the sizes of the validation, test, and meta subsets at 10%, 30%, and 10% of the training size, respectively. For the training subset, we consider two different sizes: $1,000$ and $10,000$ data points.

9.2 Datasets

In this section, ten distinct datasets are summarized, detailed in Tab.S2. To comprehensively evaluate performance across various types of datasets, we implement datasets involving three categories: tabular, textual, and image.

9.3 Hyperparameters for existing methods

In this section, we explore the impact of hyperparameters for marginal contribution-based data valuation methods.

- For LOO [16], we do not need to set arbitrary parameters, maintaining the default parameter values. For the utility function, we choose the test accuracy of a base model trained on the training subset.
- For Data Shapley [18] and Beta Shapley [19], we use a Monte Carlo method to approximate the value of data points. Specifically, the process begins by estimating the marginal contributions of data points. Following this, the Shapley value is computed based on these marginal contributions, serving as the data's assigned value. Accordingly, we configure the independent Monte Carlo chains to 10, the stopping threshold to 1.05, the sampling epoch to 100, and the minimum training set cardinality to 5.
- For DataBanzhaf [20], we set the number of utility functions to be $1,000$. Moreover, We use a two-layer MLP with 256 neurons in the hidden layer for larger datasets and 100 neurons for smaller ones.
- For InfluenceFunction [21], we also set the number of utility functions to be $1,000$. Subsequently, the cardinality of each subset is set to 0.7, indicating that 70% of it consists of data points to be evaluated.
- For KNN Shapley [22], we set the number of nearest neighbors to be equal to the size of the validation set. This is the only parameter that requires setting.
- For AME [40], we set the number of utility functions to be $1,000$. We consider the same uniform distribution for constructing subsets. For each $p \in \{0.2, 0.4, 0.6, 0.8\}$, we randomly generate 250 subsets such that the probability that a datum is included in the subset is p. As for the Lasso model, we optimize the regularization parameter using 'LassoCV' in 'scikit-learn' with its default parameter values.
- For Data-OOB [25], we set the number of weak classifiers to $1,000$, corresponding to utility function. These weak classifiers are a random forest model with decision trees, and the parameters are set to the default values in 'scikit-learn'.
- Our method, we set the size of the meta-data set to be equal to the size of the validation set. The meta network's hidden layer size is set at 10% of the meta-data size. For the training parameters, we set the max epochs to 50. Both base optimization and meta optimization use Adam optimizer, with the initial learning rate for out-optimization set at 0.01 and for in-optimization at 0.001. For base optimization, upon reaching 60% of the maximum epochs, the

Table S2: A summary of various classification datasets used in our experiments.

Dataset	Sample Size	Input Dimension	Number of Classes	Minor Class Proportion	Data Type	Source
2dplanes	40768	10	2	0.499	Tabular	[41]
electricity	38474	6	2	0.5	Tabular	[42]
fired	40768	10	2	0.498	Tabular	[41]
pol	15000	48	2	0.336	Tabular	[41]
nomao	34465	89	2	0.285	Tabular	[43]
MiniBooNE	72998	50	2	0.5	Tabular	[44]
bbc	2225	768	5	0.17	Text	[45]
IMDB	50000	768	2	0.5	Text	[46]
STL10	5000	96	10	0.01	Image	[47]
CIFAR10	50000	2048	10	0.1	Image	[48]

learning rate decays to 10% of its initial value, and at 80% of the maximum epochs, it further decays by 10%. For meta optimization, only one epoch of training is required. To maintain the original training step size, weights of all examples in a training batch are normalized to sum up to one, enforcing the constraint $|\mathcal{V}(\ell;\ \theta)| = 1$, and the normalized weight

$$\eta_i^k = \frac{\mathcal{V}(\Phi_i(\cdot, \psi_i);\ \theta^k)}{\sum_j \mathcal{V}(\Phi_j(\cdot, \psi_j);\ \theta^k) + \delta(\sum_i \mathcal{V}(\Phi_j(\cdot, \psi_i);\ \theta^k))}.$$

where $\delta(\sum_i \mathcal{V}(\cdot;\ \theta^k))$, set to $\tau > 0$ if $\sum_i \mathcal{V}(\cdot;\ \theta^k) = 0$ and 0 otherwise, prevents degeneration of $\mathcal{V}(\cdot;\ \theta^k)$ to zeros in a mini-batch, stabilizing the meta weight learning rate when used with batch normalization.

9.4 Pseudo algorithm

We provide a pseudo algorithm in Alg.S3 to illustrate the process of NDDV. It is evident that the implementation of NDDV is straightforward and easy to implement.

Algorithm S3 Pseudo-code of NDDV training

```
Input: Training data $\mathcal{D}$, meta-data set $\mathcal{D}^{'}$, batch size $n, m$, max iterations $K$.
Output: The value of data points: $\phi(x_i, y_i;\ U(S))$.
Initialize The base optimization parameter $\psi^0$ and the meta optimization parameter $\theta^0$.
for $k = 0$ to $K - 1$ do
    $\{x, y\} \leftarrow$ SampleMiniBatch($\mathcal{D}, n$).
    $\{x^{'}, y^{'}\} \leftarrow$ SampleMiniBatch($\mathcal{D}^{'}, m$).
    Formulate the base training function $\hat{\psi}^k(\theta)$ by
        $\hat{\psi}^k = \psi^k + \frac{\alpha}{N}\sum_{i=1}^{N}\sum_{t=0}^{T-1} \nabla_\psi \mathcal{H}_i(t, X_{i,t}^k, Y_{i,t}^k, \mu_t^k, \psi, \mathcal{V}(\Phi_i(X_{i,T}^k, \mu_{i,T}^k, \psi_T^k);\ \theta))\big|_{\psi^k},$
    Update the base optimization parameters $\theta^{k+1}$ by
        $\theta^{k+1} = \theta^k - \frac{\beta}{M}\sum_{i=1}^{M} \nabla_\theta \ell_i(\hat{\psi}^k(\theta))\big|_\theta^k,$
    Update the meta optimization parameters $\psi^{k+1}$ by
        $\psi^{k+1} = \psi^k + \frac{\alpha}{N}\sum_{i=1}^{N}\sum_{t=0}^{T-1} \nabla_\psi \mathcal{H}_i(t, X_{i,t}^k, Y_{i,t}^k, \mu_t^k, \psi^k, \mathcal{V}(\Phi_i(X_{i,T}^k, \mu_{i,T}^k, \psi_T^k);\ \theta^{k+1}))\big|_{\psi^k},$
    Update the weighted mean-field state $\mu_t^{k+1}$ by
        $\mu_t^{k+1} = \frac{1}{N}\sum_{i=1}^{N}\sum_{t=0}^{T-1} \mathcal{V}(\Phi(X_{i,T}^{k+1}, \mu_{i,T}^{k+1}, \psi_T^{k+1});\ \theta^{k+1}) X_{i,t}^{k+1}.$
end for
Compute the data state utility function $U_i(S)$ by
    $U_i(S) = -X_{i,T} \cdot Y_{i,T},$
Compute the dynamic marginal contribution $\Delta(x_i, y_i;\ U_i)$ by
    $\Delta(x_i, y_i;\ U_i(S)) = U_i(S) - \sum_{j \in \{1,\dots,N\}\setminus i} \frac{U_j(S)}{N-1},$
Compute the value of data points $\phi(x_i, y_i;\ U(S))$ by
    $\phi(x_i, y_i;\ U_i(S)) = \Delta(x_i, y_i;\ U_i(S)).$
```

9.5 Convergence verification for the training loss

To validate the convergence results obtained in Theorem 1 and 2 in the paper, we plot the changing tendency curves of training and meta losses with the number of epochs in our experiments, as shown in Fig.S1. The convergence tendency can be easily observed in the figures, substantiating the properness of the theoretical results in proposed theorems.

9.6 Additional results on efficiency and effectiveness

In this section, we present a comprehensive set of additional results that demonstrate the efficiency and effectiveness of NDDV. These results further substantiate our claims about the method's superior performance and practical applicability in various scenarios.

First, to evaluate the ability of different data valuation methods to detect corrupted data, we synthesize label noise on the training datasets by introducing perturbations. Specifically, we randomly select a subset of the training data points and flip their labels, simulating a label noise rate of 10%. The training sample size is set to $n \in \{1,000, 10,000\}$. However, due to the low computational efficiency of LOO, DataShapley, BetaShapley, and DataBanzhaf, we compute their results only for the smaller sample size of $n = 1,000$. After conducting data valuation, we use the k-means clustering algorithm to divide the value of data points into two clusters based on the mean. The cluster with the lower mean is considered to be the identification of corrupted data points. As shown in Tab.S3, the F1-scores for various data valuation methods are presented, showcasing their performance under the influence of mislabeled data points across six datasets. Overall, NDDV outperforms the existing methods, demonstrating its superiority in detecting corrupted data points.

Then, we analyze additional results from data valuation experiments conducted on six datasets, each with a 10% label noise rate, as shown in Fig.S2. These experiments include removing low-value data (see Fig.S2a) and adding high-value data (see Fig.S2b). Further, Fig.S2a) depicts the impact of removing the least valuable data points on performance by different data valuation methods. The results show that removing the least valuable data points has a less significant impact on performance compared to removing the most valuable data points, and NDDV's effectiveness in removing the least valuable data points is comparable to that of most existing methods. Data-OOB shows the worst performance in removing low-value data points, indicating a lack of sensitivity towards low-quality data. Similarly, Fig.S2b demonstrates the test accuracy curves of different data valuation methods in the data point addition experiment. The experiments show that adding high-value data points boosts test accuracy above the random baseline. This observation highlights the effectiveness of the data valuation methods in identifying the impact of adding high-value data points. It is apparent that both types of experiments have a modest impact on the model performance relative to existing methods.

Furthermore, we explore the impact of introducing various levels of label noise, using the pol dataset as an example. Here, we consider the six different levels of label noise rate $p_{\text{noise}} \in \{5\%, 10\%, 20\%, 30\%, 40\%, 45\%\}$. As shown in Tab.S4, NDDV consistently achieves excellent performance across all levels of label noise. Additionally, the performance of this method initially increases and then decreases with rising label noise rates, reaching its peak F1-score at a 30% noise rate. Under the influence of label noise, compared to existing methods, NDDV excels in data valuation tasks such as detecting corrupted data (see Fig.S3a), removing high/low-value data points (see Fig.S4b-S5b), and adding high/low-value data points (see Fig.S6b-S7b). Overall, its performance is comparable to that of Data-OOB.

Similarly, we investigate the effects of introducing various degrees of feature noise. In this analysis, we evaluate six distinct levels of feature noise rate, following the format described previously. As indicated in Tab.S5, NDDV attains optimal performance at all levels of label noise rate. Furthermore, as the label noise rate increases, the performance of this method generally shows a rising trend, a pattern also observed in existing methods. This demonstrates that increasing feature noise can enhance model performance, serving as a form of data augmentation. With increasing feature noise, NDDV excels in the aforementioned data valuation tasks, demonstrating overall performance comparable to that of Data-OOB and KNN-Shapley (see Fig.S3b, S4b, S5.b, S6.b, and S7.b). Particularly, in experiments involving the addition of high-value data points (see Fig.S6b), the performance improvements with NDDV are most pronounced.

9.7 Additional results on the data valuation process

In this section, we present additional results concerning the process of NDDV. Here, we consider six levels of label and feature noise rate, consistent with the settings described in Section 9.6. Fig.S8 demonstrates that with increasing label noise rates, the lower-valued data state trajectories tend to cluster, while Fig.S9 reveals that the segments of higher and lower values within the data state trajectories gradually separate with rising the label noise rate. Furthermore, as shown in Fig.S10, the data value trajectories evolve over time from concentrated to dispersed, effectively displaying the high and low values of data points. As the label noise rate increases, the learned data value trajectories become more diffuse (see Fig.S10a). These results indicate that increasing label noise enhances the visibility of individual data point values. In contrast, increasing feature noise has a smaller impact on the data value trajectories, slightly reducing the prevalence of low-value trajectories (see Fig.S10b).

9.8 Additional results on interpretability

In this section, we present additional results that enhance the interpretability of NDDV, encompassing analyses related to noise impacts and data valuation processes. Specifically, as shown in Fig.S11, the interpretable NDDV demonstrates superior F1-scores compared to existing methods across various levels of label and feature noise rate on the six datasets. Furthermore, the interpretable NDDV exhibits robust performance with label noise (see Fig.S11a) and demonstrates linear performance improvements as the feature noise rate increases (see Fig.S11b). Among existing methods, AME exhibits the poorest performance, while DataShapley and BetaShapley show instability. The performance of other methods tends to cluster more tightly.

Additionally, we demonstrate the data valuation process of the interpretable NDDV on the six datasets, including the learning of data state trajectories (see Fig.S12), data co-state trajectories (see Fig.S13), and data value trajectories (see Fig.S14). The results reveal a clear valuation process whereby, influenced by the data state and co-state trajectories, the data value trajectories evolve over time from concentrated to dispersed, effectively distinguishing between higher and lower value rankings. Consequently, our proposed method validates its effectiveness as an interpretable model.

9.9 Ablation study

We perform an ablation study on the hyperparameters in the proposed NDDV method, where we provide insights on the impact of setting changes. We use the mislabeled detection use case and the plc dataset as an example setting for the ablation study.

For all the experiments in the main text, the impact of re-weighting data points on NDDV is initially explored. As shown in Fig.S15a, NDDV enhanced with data point re-weighting demonstrates superior performance in detecting mislabeled data and removing or adding data points. Additionally, in the same experiments, Fig.S15b shows the impact of mean-field interactions parameter a at $1, 3, 5, 10$. It is observed that when $a = 10$, NDDV exhibits poorer model performance, whereas, for the other values, it shows similar performance. Furthermore, we explore the impact of the diffusion constant σ on NDDV. Specifically, we set σ at $0.001, 0.01, 0.1, 1.0$ to compare model performance. As shown in Fig.S15c, the model performance significantly deteriorates at $\sigma = 1.0$. It can be observed that excessively high values of σ increase the random dynamics of the data points, causing the model performance to randomness. Then, we use the meta-dataset of size identical to the size of the validation set. Naturally, we want to examine the effect of the size of the metadata set on the detection rate of mislabeled data. We illustrate the performance of the detection rate with different metadata sizes: 10, 100, and 300. Fig.S15d shows that various metadata sizes have a minimal impact on model performance. Finally, we analyzed the impact of the meta-hidden points on model performance. As shown in Fig.S15e, it is an event in which the meta hidden points are set to 5, and there is a notable decline in the performance of NDDV. In fact, smaller meta-hidden points tend to lead to underfitting, making it challenging for the model to achieve optimal performance.

References

[1] Weinan, E.: A proposal on machine learning via dynamical systems. Communications in Mathematics and Statistics **1**(5), 1–11 (2017)

[2] He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)

[3] Lu, Y., Zhong, A., Li, Q., Dong, B.: Beyond finite layer neural networks: Bridging deep architectures and numerical differential equations. In: International Conference on Machine Learning, pp. 3276–3285 (2018). PMLR

[4] Sonoda, S., Murata, N.: Transport analysis of infinitely deep neural network. Journal of Machine Learning Research **20**(2), 1–52 (2019)

[5] Hu, B., Lessard, L.: Control interpretations for first-order optimization methods. In: 2017 American Control Conference (ACC), pp. 3114–3119 (2017). IEEE

[6] Han, J., Li, Q., *et al.*: A mean-field optimal control formulation of deep learning. Research in the Mathematical Sciences **6**(1), 1–41 (2019)

[7] Li, Q., Chen, L., Tai, C., Weinan, E.: Maximum principle based algorithms for deep learning. Journal of Machine Learning Research **18**(165), 1–29 (2018)

[8] Li, Q., Hao, S.: An optimal control approach to deep learning and applications to discrete-weight neural networks. In: International Conference on Machine Learning, pp. 2985–2994 (2018). PMLR

[9] Zhang, D., Zhang, T., Lu, Y., Zhu, Z., Dong, B.: You only propagate once: Accelerating adversarial training via maximal principle. Advances in neural information processing systems **32** (2019)

[10] Pavliotis, G.A.: Stochastic Processes and Applications: Diffusion Processes, the Fokker-Planck and Langevin Equations vol. 60. Springer, ??? (2014)

[11] Simsekli, U., Sagun, L., Gurbuzbalaban, M.: A tail-index analysis of stochastic gradient noise in deep neural networks. In: International Conference on Machine Learning, pp. 5827–5837 (2019). PMLR

[12] Du, S.S., Zhai, X., Poczos, B., Singh, A.: Gradient descent provably optimizes over-parameterized neural networks. arXiv preprint arXiv:1810.02054 (2018)

[13] Du, S., Lee, J., Li, H., Wang, L., Zhai, X.: Gradient descent finds global minima of deep neural networks. In: International Conference on Machine Learning, pp. 1675–1685 (2019). PMLR

[14] Li, Q., Tai, C., Weinan, E.: Stochastic modified equations and adaptive stochastic gradient algorithms. In: International Conference on Machine Learning, pp. 2101–2110 (2017). PMLR

[15] An, J., Lu, J., Ying, L.: Stochastic modified equations for the asynchronous stochastic gradient descent. Information and Inference: A Journal of the IMA **9**(4), 851–873 (2020)

[16] Koh, P.W., Liang, P.: Understanding black-box predictions via influence functions. In: International Conference on Machine Learning, pp. 1885–1894 (2017). PMLR

[17] Jia, R., Dao, D., Wang, B., Hubis, F.A., Hynes, N., Gürel, N.M., Li, B., Zhang, C., Song, D., Spanos, C.J.: Towards efficient data valuation based on the shapley value. In: The 22nd

International Conference on Artificial Intelligence and Statistics, pp. 1167–1176 (2019). PMLR

[18] Ghorbani, A., Zou, J.: Data shapley: Equitable valuation of data for machine learning. In: International Conference on Machine Learning, pp. 2242–2251 (2019). PMLR

[19] Kwon, Y., Zou, J.: Beta shapley: a unified and noise-reduced data valuation framework for machine learning. arXiv preprint arXiv:2110.14049 (2021)

[20] Wang, T., Jia, R.: Data banzhaf: A data valuation framework with maximal robustness to learning stochasticity. arXiv preprint arXiv:2205.15466 (2022)

[21] Feldman, V., Zhang, C.: What neural networks memorize and why: Discovering the long tail via influence estimation. Advances in Neural Information Processing Systems **33**, 2881–2891 (2020)

[22] Jia, R., Dao, D., Wang, B., Hubis, F.A., Gurel, N.M., Li, B., Zhang, C., Spanos, C.J., Song, D.: Efficient task-specific data valuation for nearest neighbor algorithms. arXiv preprint arXiv:1908.08619 (2019)

[23] Xu, X., Wu, Z., Foo, C.S., Low, B.K.H.: Validation free and replication robust volume-based data valuation. Advances in Neural Information Processing Systems **34**, 10837–10848 (2021)

[24] Just, H.A., Kang, F., Wang, J.T., Zeng, Y., Ko, M., Jin, M., Jia, R.: Lava: Data valuation without pre-specified learning algorithms. arXiv preprint arXiv:2305.00054 (2023)

[25] Kwon, Y., Zou, J.: Data-oob: out-of-bag estimate as a simple and efficient data value. In: International Conference on Machine Learning, pp. 18135–18152 (2023). PMLR

[26] Lundberg, S.M., Lee, S.-I.: A unified approach to interpreting model predictions. Advances in neural information processing systems **30** (2017)

[27] Covert, I., Lundberg, S., Lee, S.-I.: Explaining by removing: A unified framework for model explanation. Journal of Machine Learning Research **22**(209), 1–90 (2021)

[28] Stier, J., Gianini, G., Granitzer, M., Ziegler, K.: Analysing neural network topologies: a game theoretic approach. Procedia Computer Science **126**, 234–243 (2018)

[29] Ghorbani, A., Zou, J.Y.: Neuron shapley: Discovering the responsible neurons. Advances in Neural Information Processing Systems **33**, 5922–5932 (2020)

[30] Sim, R.H.L., Zhang, Y., Chan, M.C., Low, B.K.H.: Collaborative machine learning with incentive-aware model rewards. In: International Conference on Machine Learning, pp. 8927–8936 (2020). PMLR

[31] Xu, X., Lyu, L., Ma, X., Miao, C., Foo, C.S., Low, B.K.H.: Gradient driven rewards to guarantee fairness in collaborative machine learning. Advances in Neural Information Processing Systems **34**, 16104–16117 (2021)

[32] Benmerzoug, A., de Benito Delgado, M.: [re] if you like shapley then you'll love the core. In: ML Reproducibility Challenge 2022 (2023)

[33] Rozemberczki, B., Watson, L., Bayer, P., Yang, H.-T., Kiss, O., Nilsson, S., Sarkar, R.: The shapley value in machine learning. arXiv preprint arXiv:2202.05594 (2022)

[34] Yoon, J., Arik, S., Pfister, T.: Data valuation using reinforcement learning. In: International Conference on Machine Learning, pp. 10842–10851 (2020). PMLR

[35] Basu, S., Pope, P., Feizi, S.: Influence functions in deep learning are fragile. arXiv preprint arXiv:2006.14651 (2020)

[36] Bachrach, Y., Markakis, E., Resnick, E., Procaccia, A.D., Rosenschein, J.S., Saberi, A.: Approximating power indices: theoretical and empirical analysis. Autonomous Agents and Multi-Agent Systems **20**(2), 105–122 (2010)

[37] Shrikumar, A., Greenside, P., Kundaje, A.: Learning important features through propagating activation differences. In: International Conference on Machine Learning, pp. 3145–3153 (2017). PMLR

[38] Ancona, M., Oztireli, C., Gross, M.: Explaining deep neural networks with a polynomial time algorithm for shapley value approximation. In: International Conference on Machine Learning, pp. 272–281 (2019). PMLR

[39] Jiang, K., Liang, W., Zou, J.Y., Kwon, Y.: Opendataval: a unified benchmark for data valuation. Advances in Neural Information Processing Systems **36** (2023)

[40] Lin, J., Zhang, A., Lécuyer, M., Li, J., Panda, A., Sen, S.: Measuring the effect of training data on deep learning predictions via randomized experiments. In: International Conference on Machine Learning, pp. 13468–13504 (2022). PMLR

[41] Feurer, M., Van Rijn, J.N., Kadra, A., Gijsbers, P., Mallik, N., Ravi, S., Müller, A., Vanschoren, J., Hutter, F.: Openml-python: an extensible python api for openml. Journal of Machine Learning Research **22**(100), 1–5 (2021)

[42] Gama, J., Medas, P., Castillo, G., Rodrigues, P.: Learning with drift detection. In: Advances in Artificial Intelligence–SBIA 2004: 17th Brazilian Symposium on Artificial Intelligence, Sao Luis, Maranhao, Brazil, September 29-Ocotber 1, 2004. Proceedings 17, pp. 286–295 (2004). Springer

[43] Candillier, L., Lemaire, V.: Design and analysis of the nomao challenge active learning in the real-world. In: Proceedings of the ALRA: Active Learning in Real-world Applications, Workshop ECML-PKDD, pp. 1–15 (2012). Citeseer

[44] Roe, B.P., Yang, H.-J., Zhu, J., Liu, Y., Stancu, I., McGregor, G.: Boosted decision trees as an alternative to artificial neural networks for particle identification. Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment **543**(2-3), 577–584 (2005)

[45] Greene, D., Cunningham, P.: Practical solutions to the problem of diagonal dominance in kernel document clustering. In: Proceedings of the 23rd International Conference on Machine Learning, pp. 377–384 (2006)

[46] Maas, A., Daly, R.E., Pham, P.T., Huang, D., Ng, A.Y., Potts, C.: Learning word vectors for sentiment analysis. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pp. 142–150 (2011)

[47] Coates, A., Ng, A., Lee, H.: An analysis of single-layer networks in unsupervised feature learning. In: Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, pp. 215–223 (2011). JMLR Workshop and Conference Proceedings

[48] Krizhevsky, A.: Learning multiple layers of features from tiny images. Master's thesis, University of Tront (2009)

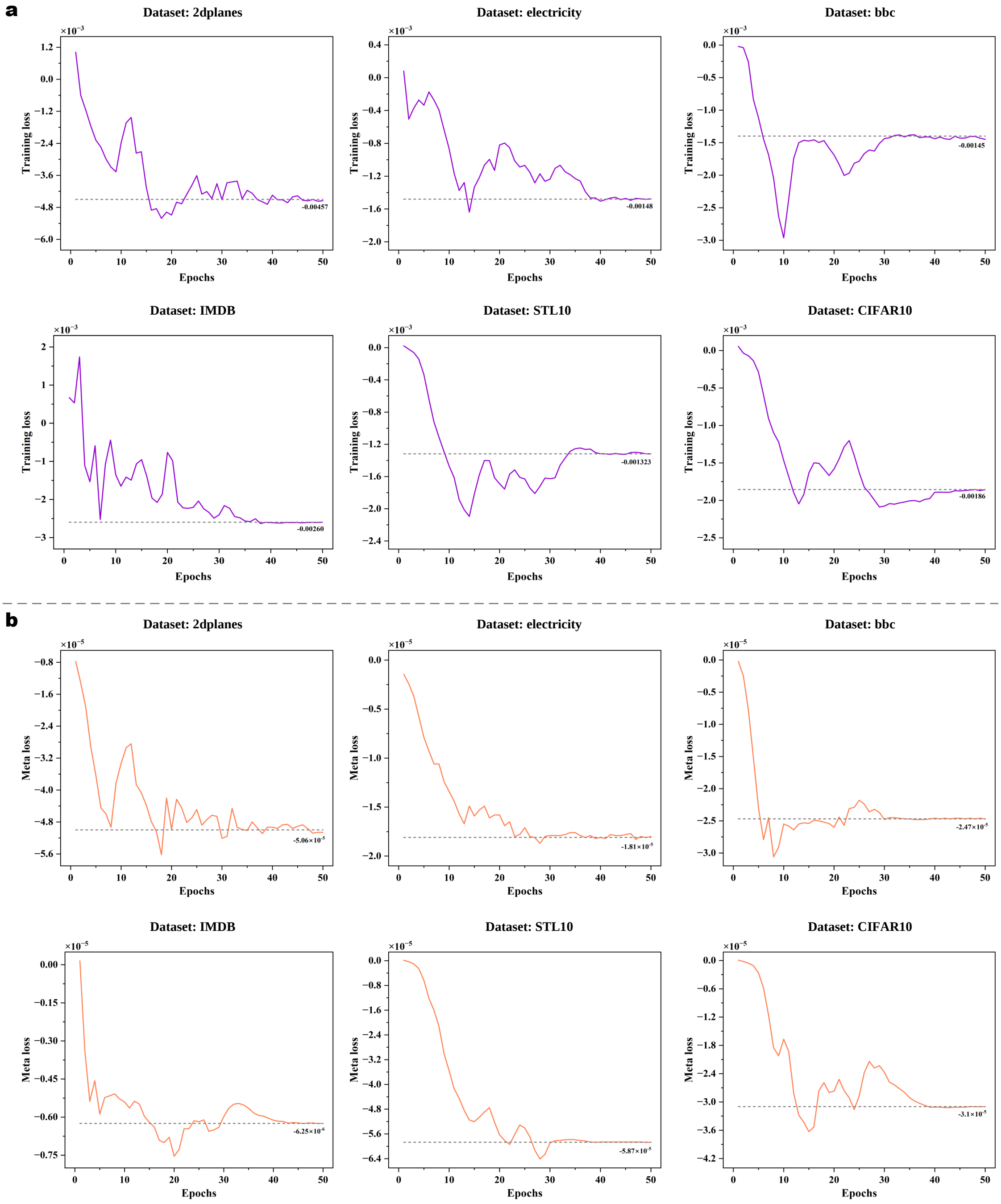

Fig. S1: **Convergence verification for NDDV. a.** Training loss tendency curves on the six datasets. **b.** Meta loss tendency curves on the six datasets.

Table S3: F1-score of different data valuation methods on the six datasets when (left) $n = 1,000$ and (right) $n = 10,000$.

Dataset				$n = 1,000$						$n = 10,000$		
	LOO	Data Shapley	Beta Shapley	Data Banzhaf	Influence Function	KNN Shapley	AME	Data -OOB	NDDV	KNN Shapley	AME	Data -OOB	NDDV
2dplanes	0.18± 0.003	0.17± 0.005	0.16± 0.003	0.16± 0.009	0.18± 0.005	0.30± 0.007	0.18± 0.009	0.46± 0.007	**0.67**± 0.005	0.37± 0.004	0.01± 0.012	0.71± 0.002	**0.79**± 0.005
electricity	0.18± 0.004	0.17± 0.004	0.19± 0.006	0.18± 0.002	0.19± 0.003	0.23± 0.006	0.01± 0.010	**0.37**± 0.002	*0.36*± 0.002	0.32± 0.001	0.01± 0.009	0.38± 0.003	**0.44**± 0.002
bbc	0.12± 0.004	0.11± 0.004	0.11± 0.003	0.18± 0.005	0.16± 0.002	0.31± 0.008	0.11± 0.009	0.18± 0.004	**0.86**± 0.002	0.52± 0.005	0.01± 0.010	0.73± 0.002	**0.85**± 0.006
IMDB	0.12± 0.002	0.09± 0.004	0.09± 0.003	0.15± 0.002	0.16± 0.009	0.22± 0.008	0.18± 0.011	0.17± 0.005	**0.27**± 0.007	0.29± 0.002	0.18± 0.012	0.48± 0.002	**0.52**± 0.003
STL10	0.13± 0.006	0.17± 0.004	0.16± 0.002	0.18± 0.005	0.14± 0.009	0.28± 0.007	0.01± 0.009	0.22± 0.003	**0.71**± 0.008	0.16± 0.009	0.01± 0.012	0.77 0.002	**0.91**± 0.003
CIFAR10	0.18± 0.004	0.19± 0.003	0.20± 0.005	0.17± 0.002	0.19± 0.007	0.24± 0.004	0.02± 0.008	0.40± 0.004	**0.59**± 0.004	0.27± 0.009	0.01± 0.010	0.46± 0.001	**0.58**± 0.004

Note: The mean and standard deviation of the F1-score are derived from 5 independent experiments. The highest and second-highest results are highlighted in bold and underlined, respectively.

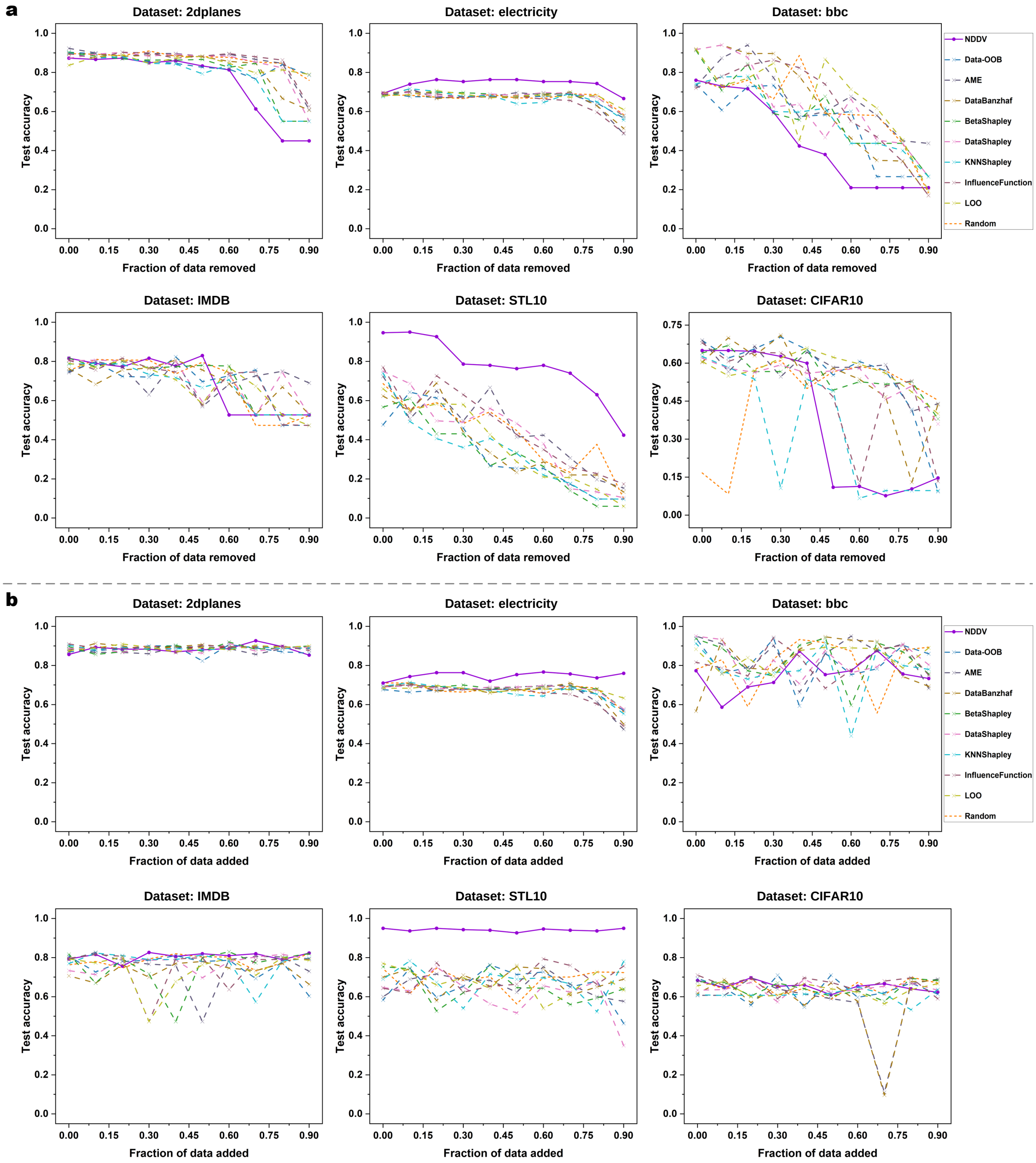

Fig. S2: **Additional results on removal and addition experiment. a.** Removing low-value data experiment on six datasets with 10% label noisy rate. Test accuracy curves show the trend after removing the least valuable data points. **b.** Adding high-value data experiment on six datasets with 10% label noisy rate. Test accuracy curves show the trend after adding the most valuable data points.

Table S4: F1-score of different data valuation methods on the different label noise rates.

Noise Rate	LOO	Data Shapley	Beta Shapley	Data Banzhaf	Influence Function	KNN Shapley	AME	Data -OOB	NDDV
5%	0.09± 0.003	0.12± 0.007	0.11± 0.008	0.09± 0.004	0.11± 0.003	0.17± 0.003	0.01± 0.009	0.62± 0.002	**0.74±** 0.003
10%	0.16± 0.007	0.19± 0.010	0.19± 0.009	0.18± 0.005	0.18± 0.003	0.30± 0.003	0.18± 0.010	0.74± 0.002	**0.76±** 0.003
20%	0.30± 0.005	0.25± 0.008	0.25± 0.008	0.31± 0.002	0.31± 0.002	0.45± 0.004	0.010± 0.009	**0.79±** 0.001	*0.77±* 0.001
30%	0.39± 0.003	0.52± 0.012	0.51± 0.010	0.42± 0.002	0.42± 0.008	0.55± 0.002	0.46± 0.011	**0.80±** 0.001	*0.78±* 0.004
40%	0.54± 0.008	0.55± 0.008	0.56± 0.008	0.48± 0.003	0.46± 0.004	0.60± 0.002	0.58± 0.010	0.73± 0.001	**0.74±** 0.002
45%	0.55± 0.007	0.55± 0.008	0.62± 0.009	0.48± 0.003	0.48± 0.001	0.56± 0.004	0.27± 0.009	0.63± 0.001	**0.67±** 0.004

Note: The mean and standard deviation of the F1-score are derived from 5 independent experiments. The highest and second-highest results are highlighted in bold and underlined, respectively.

Table S5: F1-score of different data valuation methods on the different feature noise rates.

Noise Rate	LOO	Data Shapley	Beta Shapley	Data Banzhaf	Influence Function	KNN Shapley	AME	Data -OOB	NDDV
5%	0.09± 0.007	0.10± 0.009	0.10± 0.007	0.07± 0.004	0.10± 0.003	0.17± 0.003	0.09± 0.012	0.15± 0.002	**0.30±** 0.006
10%	0.18± 0.007	0.18± 0.010	0.18± 0.009	0.15± 0.005	0.15± 0.003	0.15± 0.003	0.18± 0.010	0.21± 0.002	**0.28±** 0.003
20%	0.33± 0.005	0.01± 0.008	0.01± 0.008	0.28± 0.002	0.30± 0.002	0.27± 0.002	0.01± 0.010	0.32± 0.001	**0.34±** 0.003
30%	0.43± 0.008	0.01± 0.012	0.01± 0.010	0.33± 0.002	0.35± 0.008	0.35± 0.002	0.01± 0.012	0.37± 0.001	**0.45±** 0.005
40%	0.51± 0.008	0.01± 0.010	0.01± 0.008	0.01± 0.003	0.37± 0.004	0.40± 0.002	0.01± 0.010	0.43± 0.001	**0.57±** 0.004
45%	0.53± 0.007	0.01± 0.011	0.01± 0.009	0.50± 0.003	0.47± 0.001	0.39± 0.002	0.01± 0.012	0.46± 0.001	**0.62±** 0.006

Note: The mean and standard deviation of the F1-score are derived from 5 independent experiments. The highest and second-highest results are highlighted in bold and underlined, respectively.

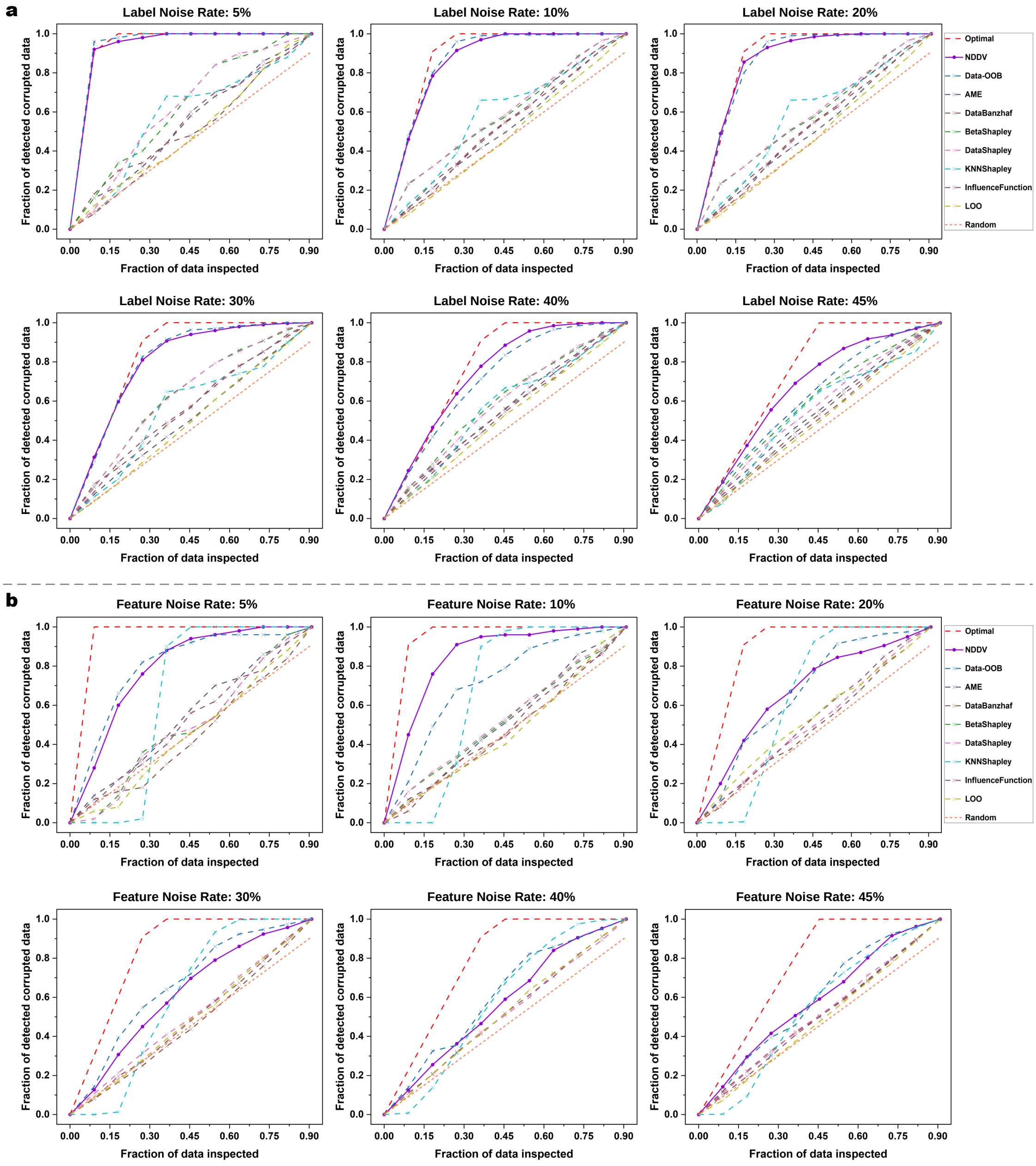

Fig. S3: **Detecting corrupted data under the influence of noise. a.** Detecting corrupted data on the six different levels of label noise rate. **b.** Detecting corrupted data on the six different levels of feature noise rate.

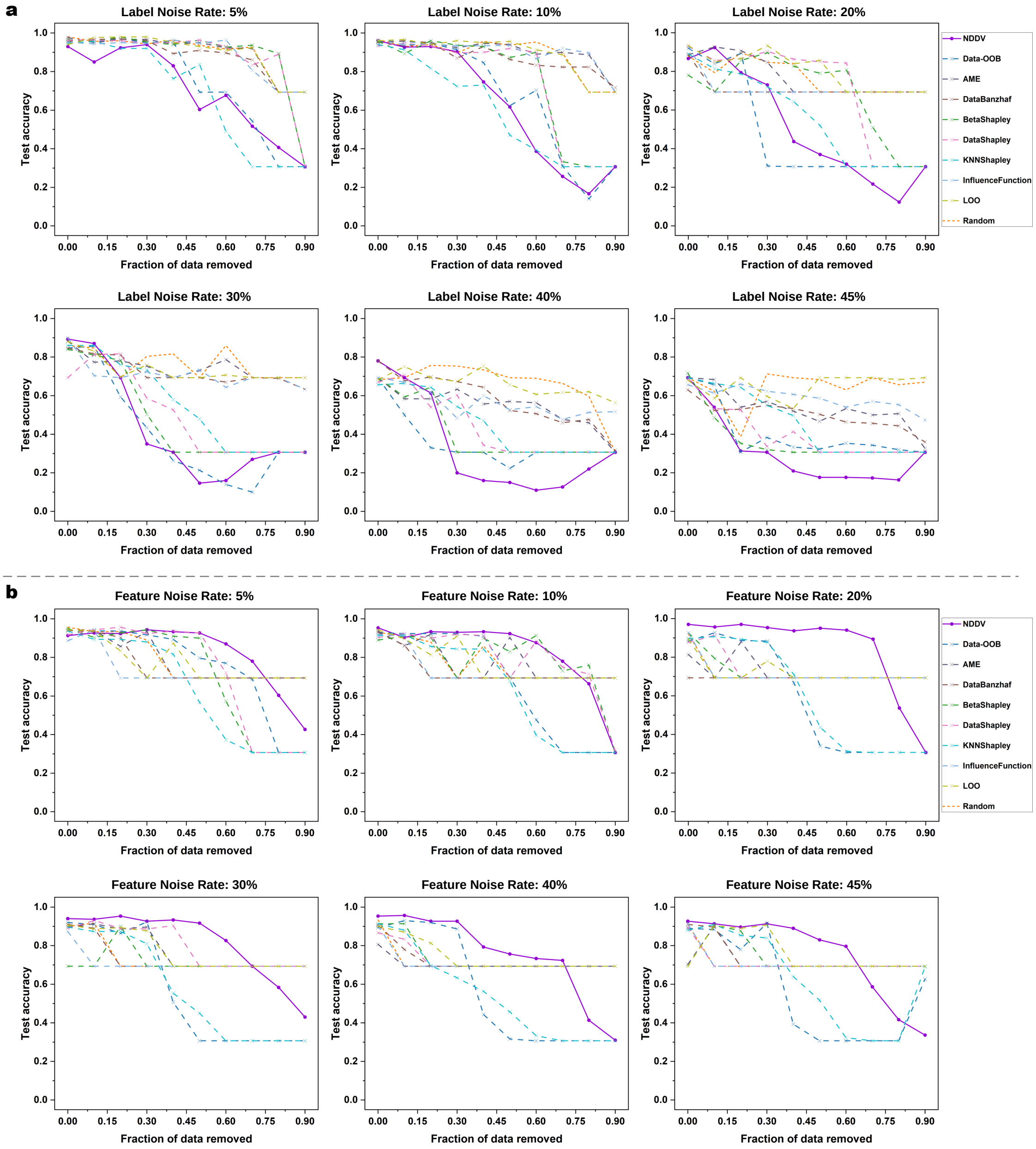

Fig. S4: **High-value data points removal experiment under the influence of noise. a.** Removing high-value data points experiment on the six different levels of label noise rate. **b.** Removing high-value data points experiment on the six different levels of feature noise rate.

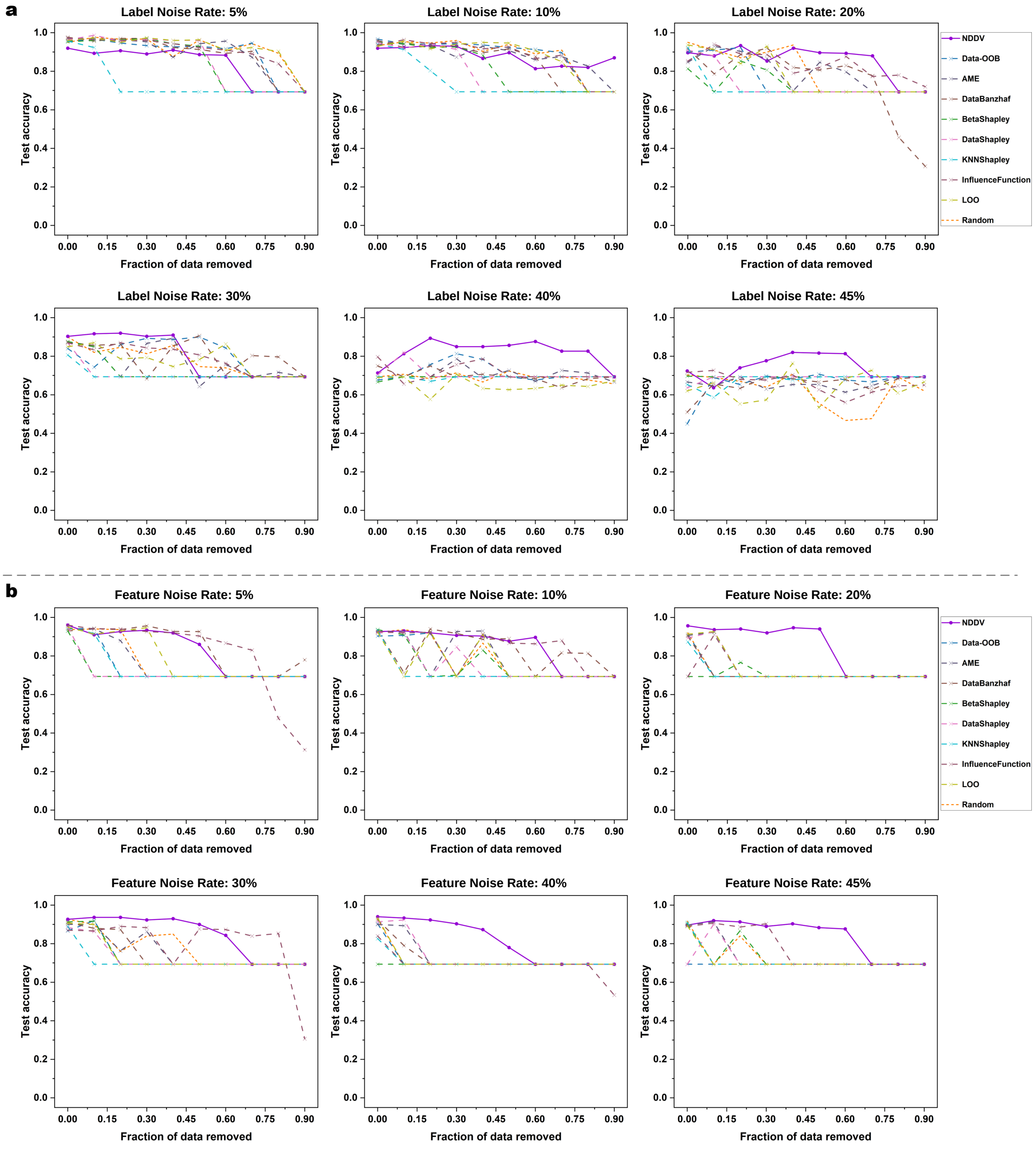

Fig. S5: **Low-value data points removal experiment under the influence of noise. a.** Removing low-value data points experiment on the six different levels of label noise rate. **b.** Removing low-value data points experiment on the six different levels of feature noise rate.

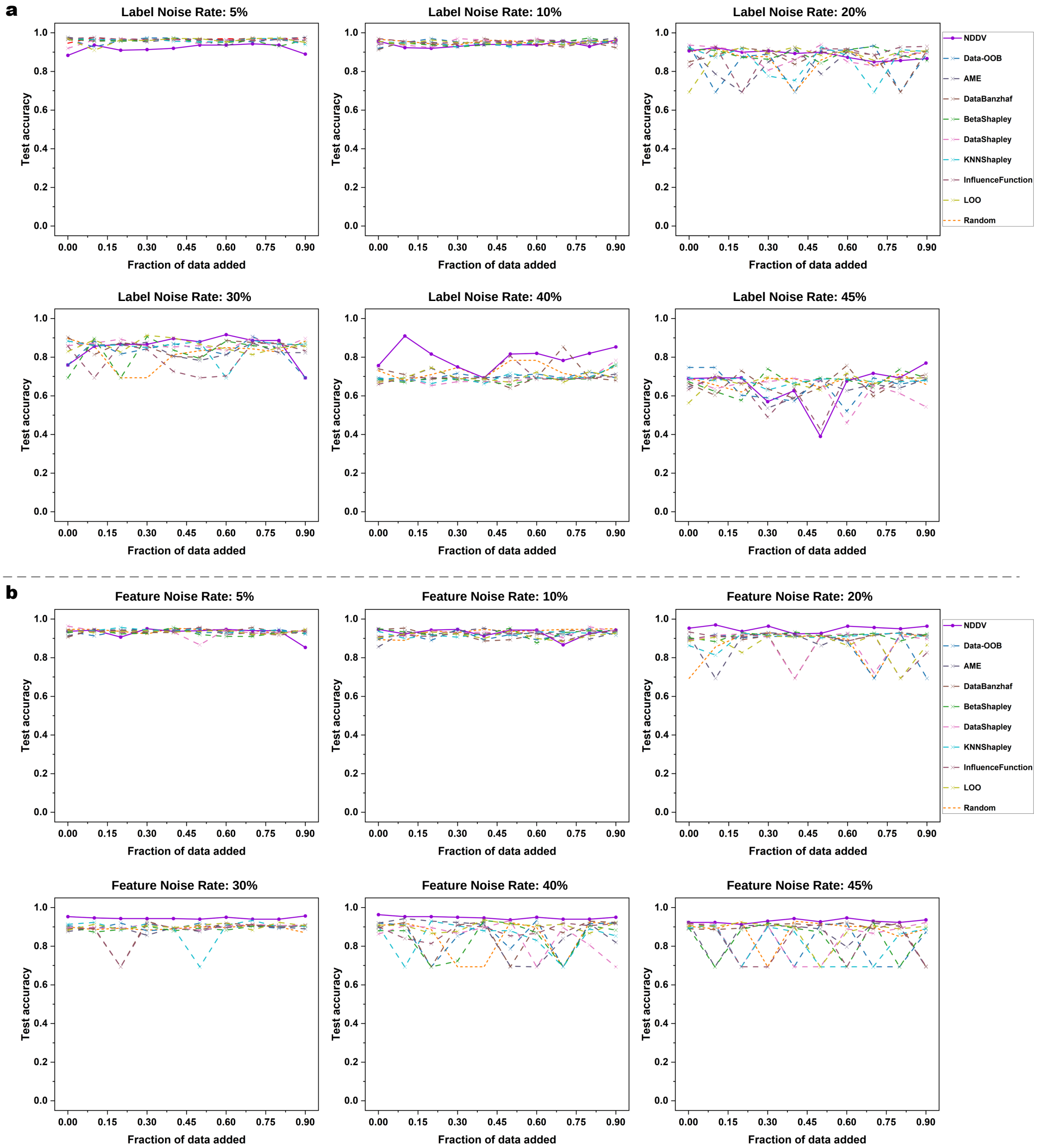

Fig. S6: **High-value data points addition experiment under the influence of noise. a.** Adding high-value data points experiment on the six different levels of label noise rate. **b.** Adding high-value data points experiments on the six different levels of feature noise rate.

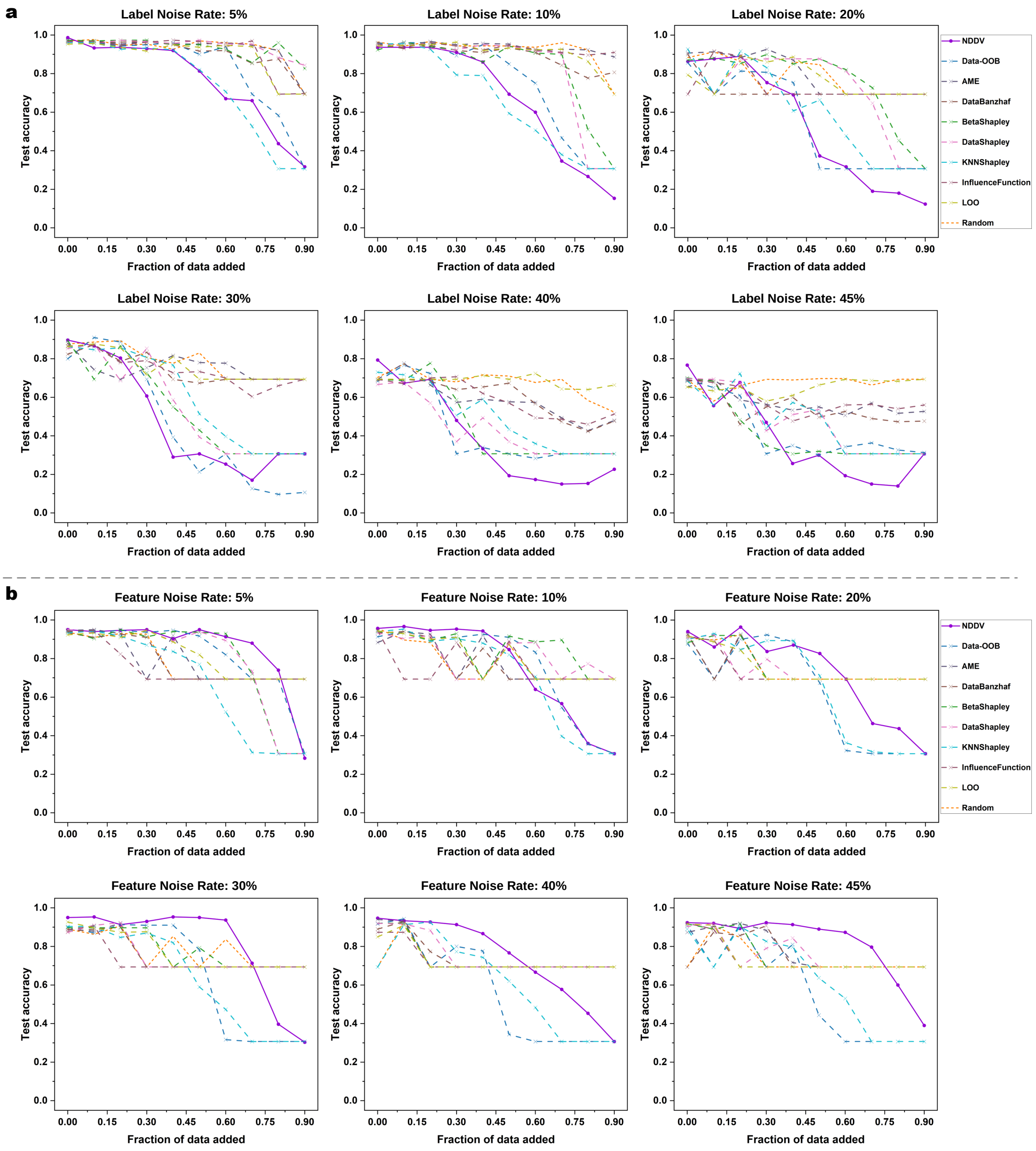

Fig. S7: **Low-value data points addition experiment under the influence of noise. a.** Adding low-value data points experiment on the six different levels of label noise rate. **b.** Adding low-value data points experiments on the six different levels of feature noise rate.

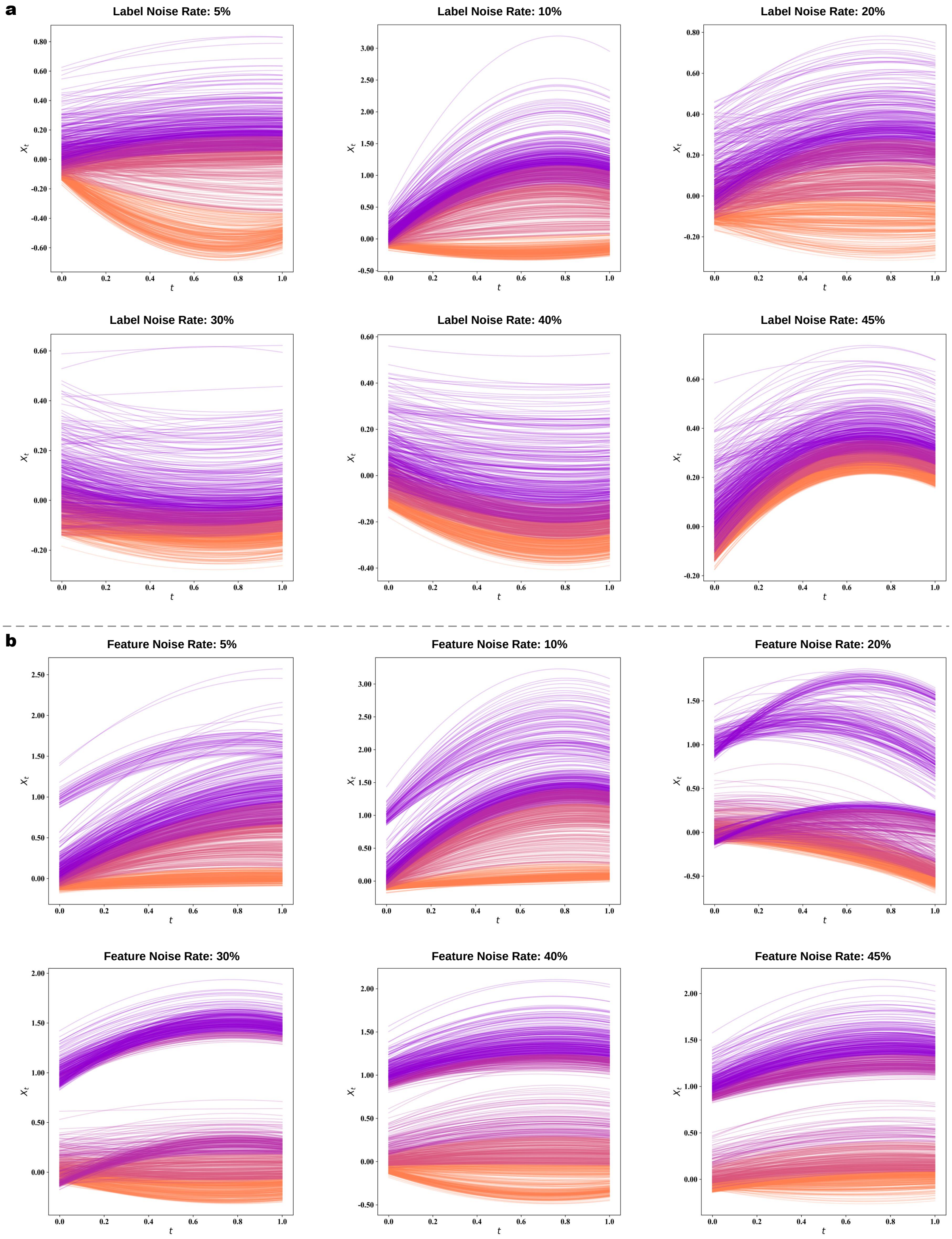

Fig. S8: **Data state trajectories for NDDV under the influence of noise. a.** Data state trajectories on the six different levels of label noise rate. **b.** Data state trajectories on the six different levels of feature noise rate.

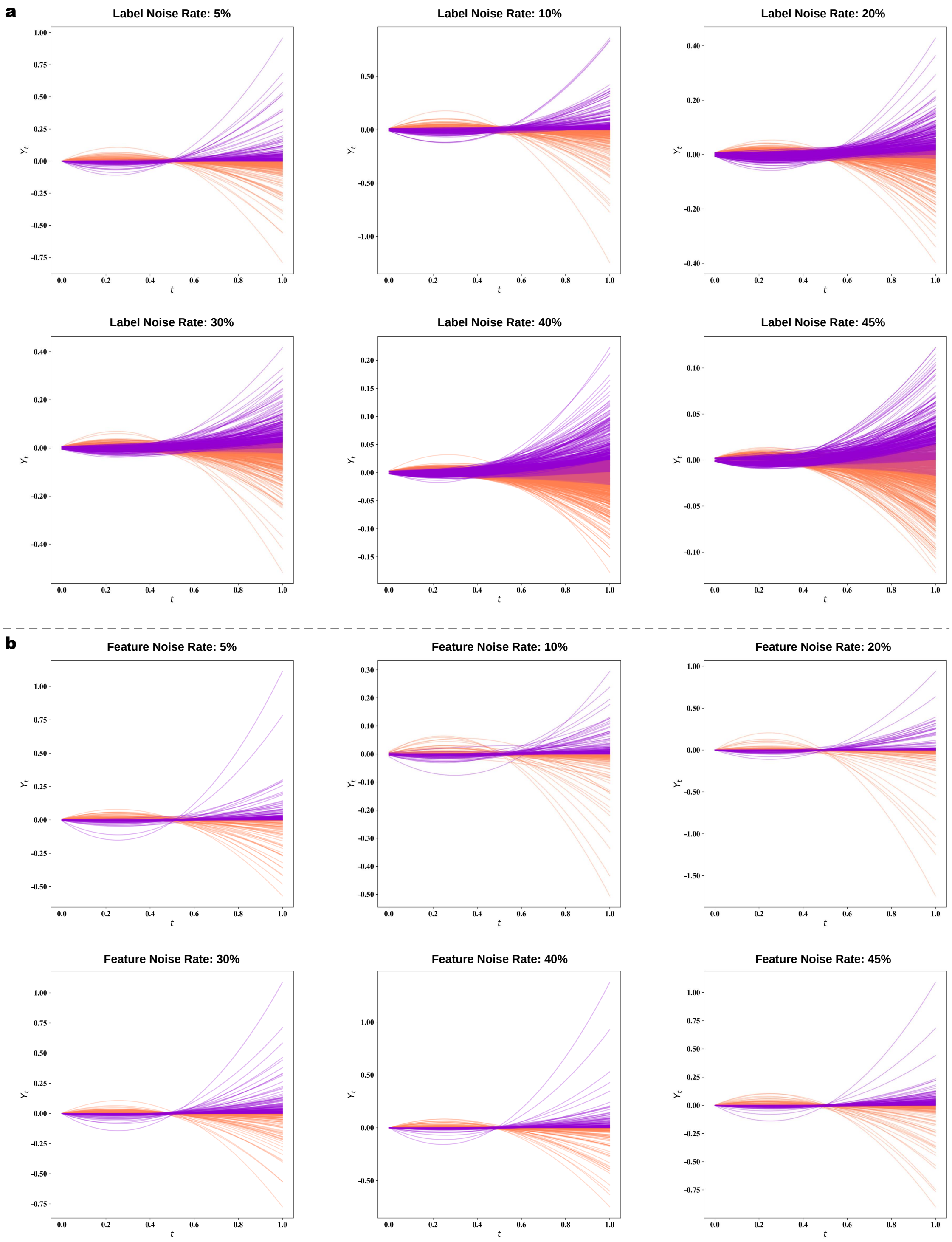

Fig. S9: **Data co-state trajectories for NDDV under the influence of noise. a.** Data co-state trajectories on the six different levels of label noise rate. **b.** Data co-state trajectories on the six different levels of feature noise rate.

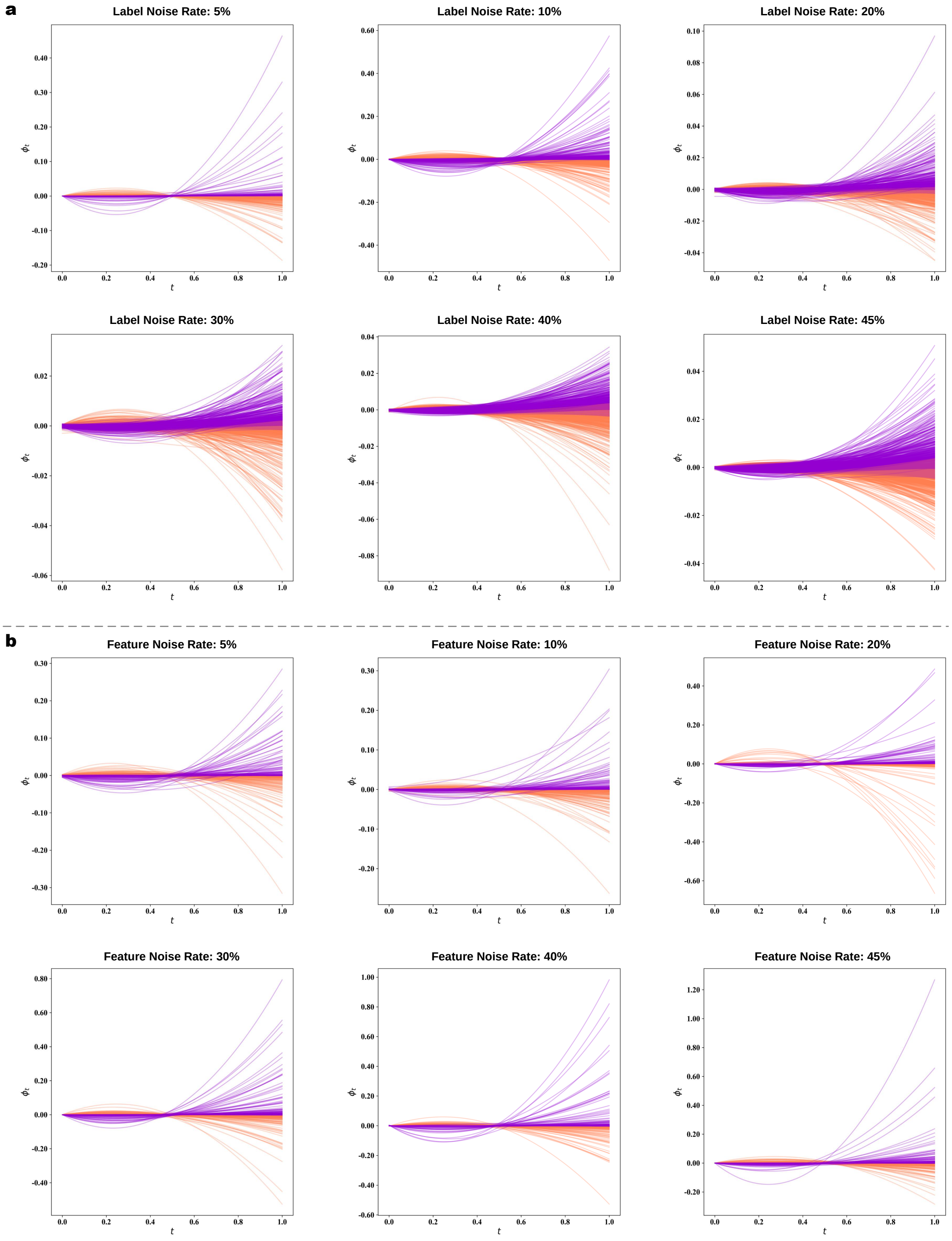

Fig. S10: **Data value trajectories for NDDV under the influence of noise. a.** Data value trajectories on the six different levels of label noise rate. **b.** Data value trajectories on the six different levels of feature noise rate.

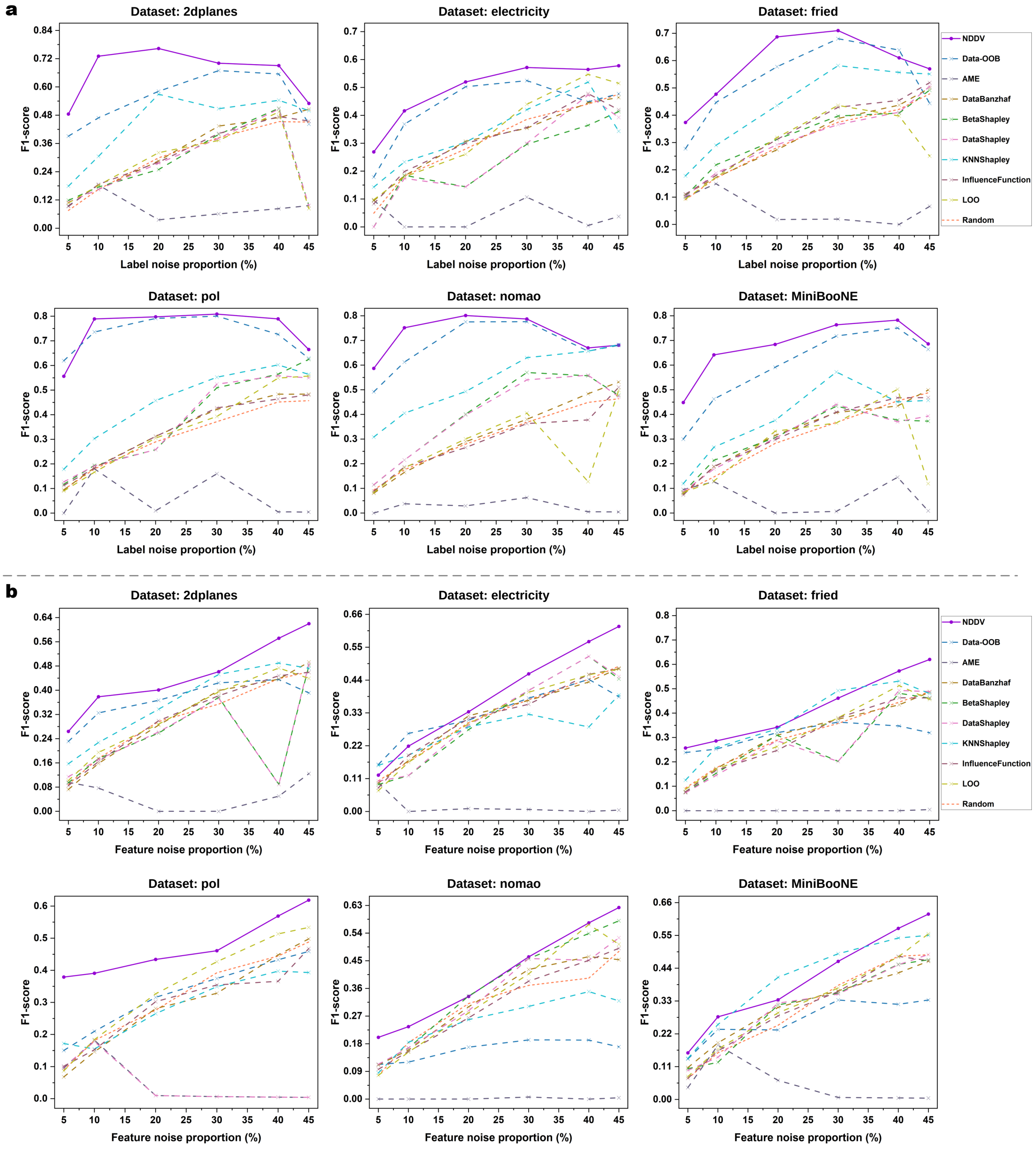

Fig. S11: **Noisy data detection for the interpretable NDDV on the six datasets. a.** Noisy label data detection. **b.** Noisy feature data detection. The F1-score of various methods is compared on the six noise proportion settings. The higher F1-score indicates superior performance.

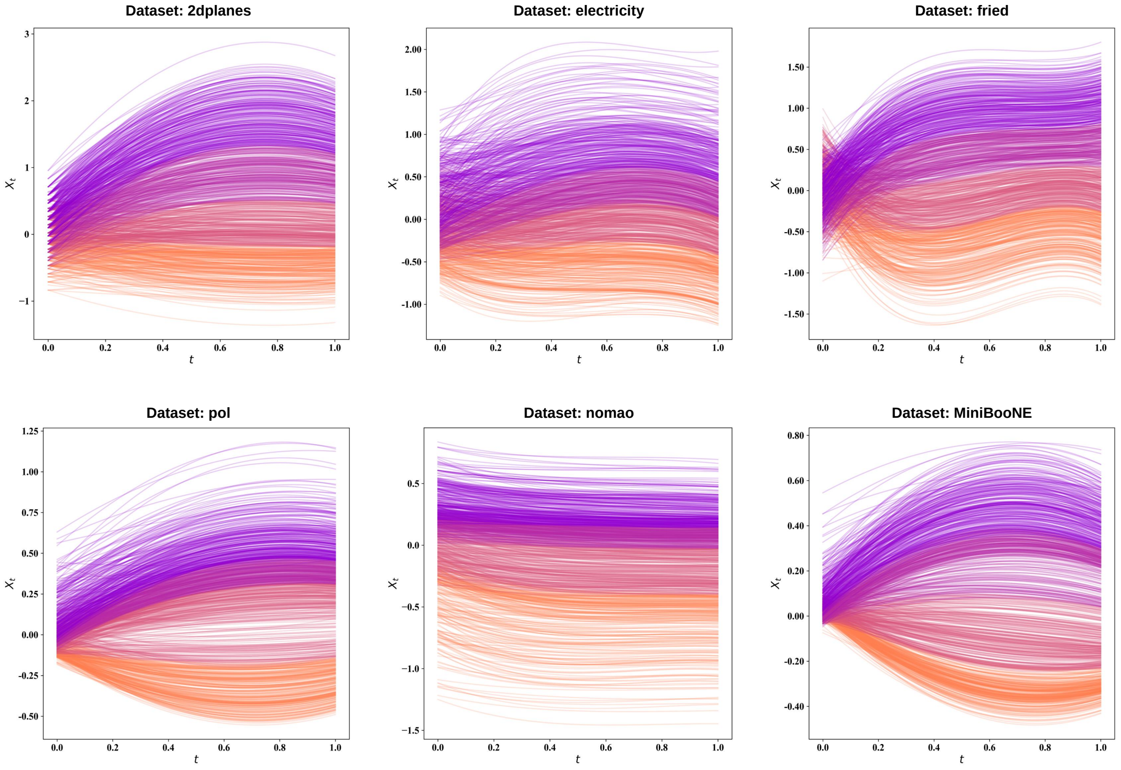

Fig. S12: **Data state trajectories for the interpretable NDDV on the six datasets.**

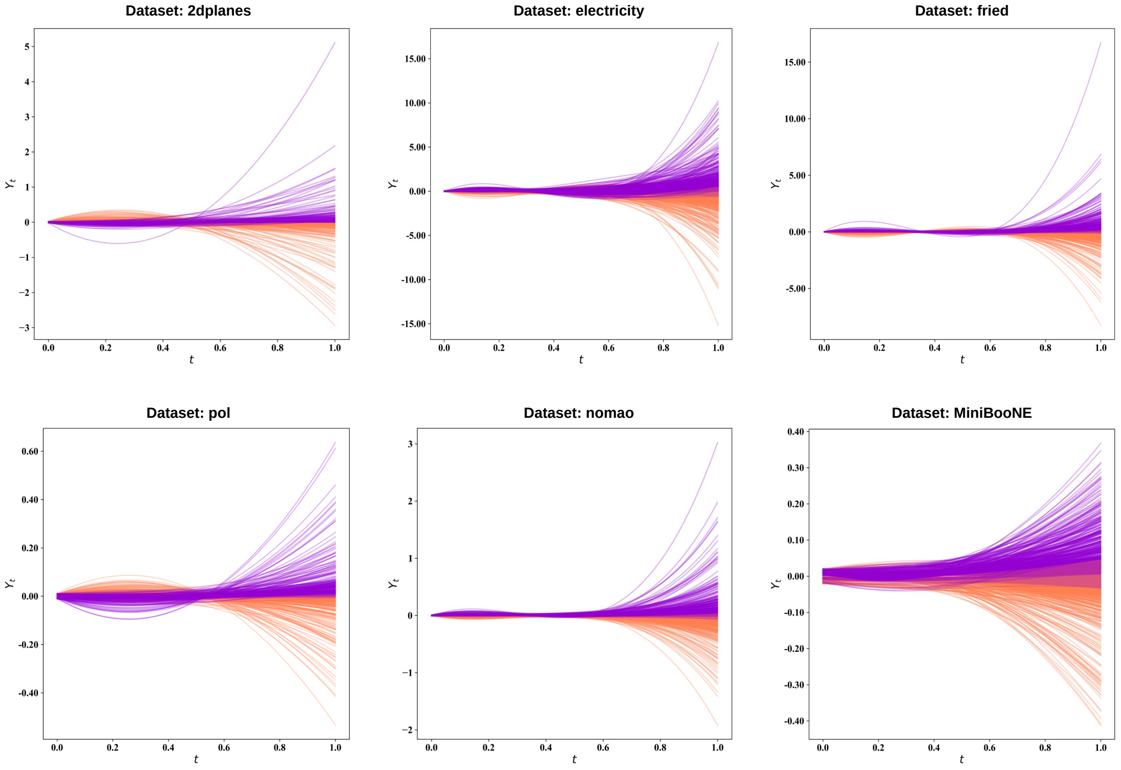

Fig. S13: **Data co-state trajectories for the interpretable NDDV on the six datasets.**

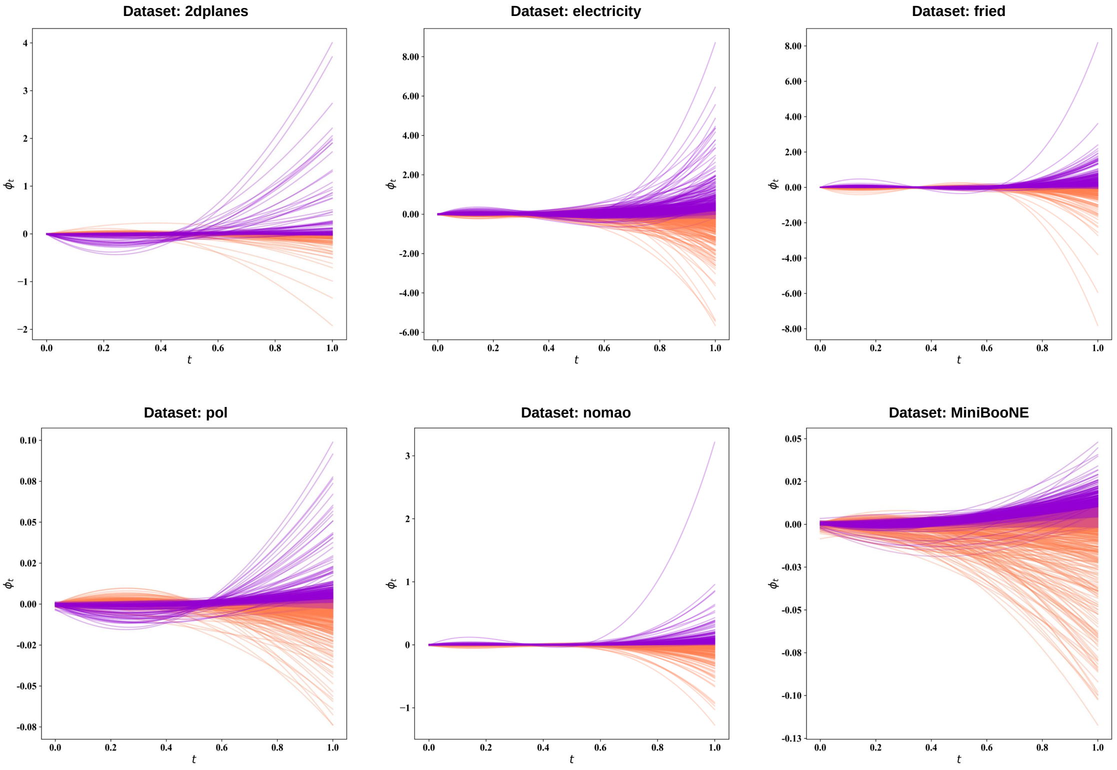

Fig. S14: **Data value trajectories for the interpretable NDDV on the six datasets.**

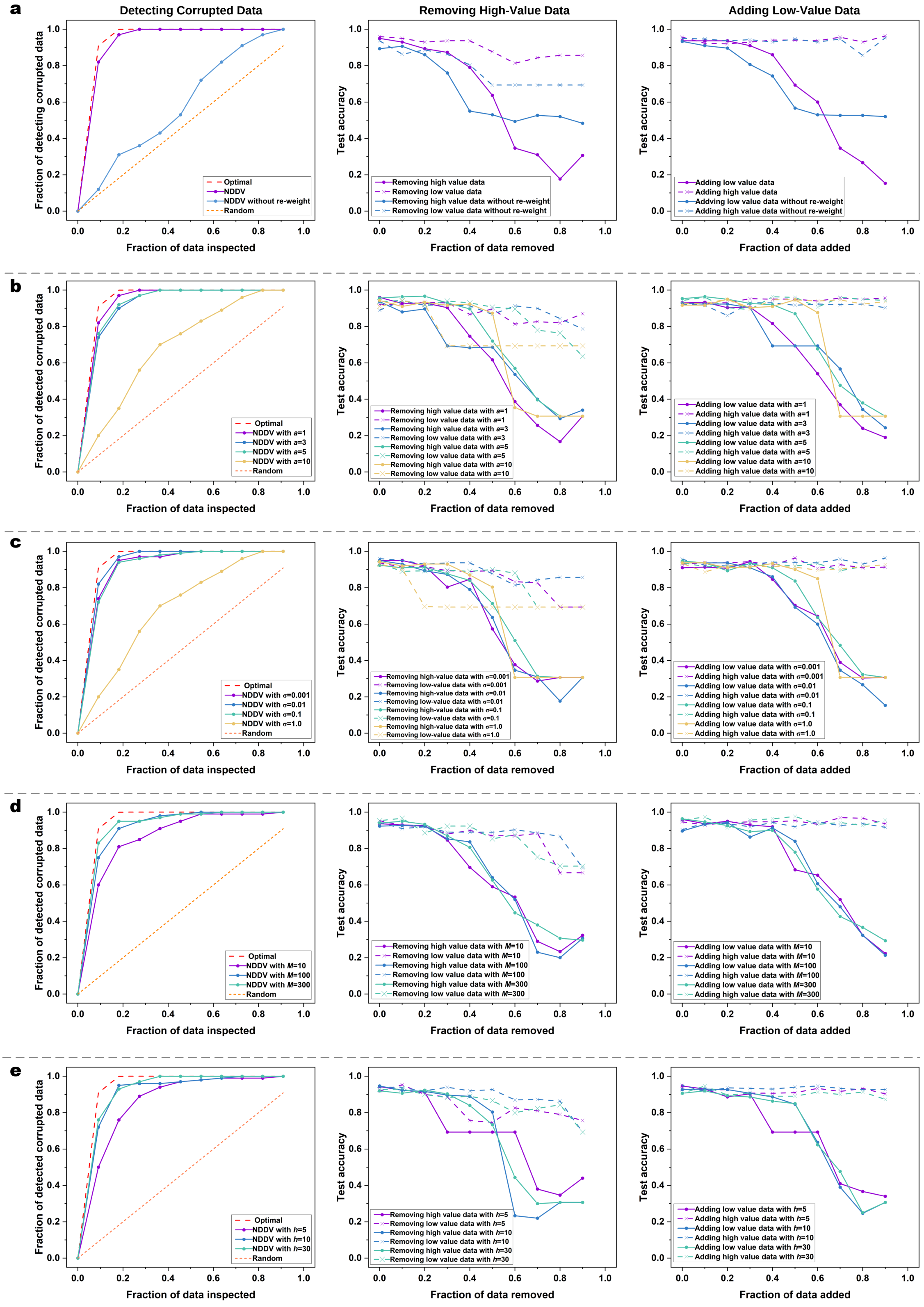

Fig. S15: **Ablation study for NDDV. a.** Impact of data points re-weighting. **b.** Impact of the mean-field interactions. **c.** Impact of the diffusion constant. **d.** Impact of the metadata sizes. **e.** Impact of the meta hidden points.