Geometrically Inspired Kernel Machines for Collaborative Learning Beyond Gradient Descent

MOHIT KUMAR*, University of Rostock, Germany and Software Competence Center Hagenberg GmbH, Austria
ALEXANDER VALENTINITSCH, Software Competence Center Hagenberg GmbH, Austria
MAGDALENA FUCHS, ETH Zürich, Switzerland
MATHIAS BRUCKER, Software Competence Center Hagenberg GmbH, Austria
JULIANA BOWLES, University of St Andrews, UK
ADNAN HUSAKOVIC, Primetals Technologies Austria GmbH, Austria
ALI ABBAS, Primetals Technologies Austria GmbH, Austria
BERNHARD A. MOSER, Johannes Kepler University, Austria and Software Competence Center Hagenberg GmbH, Austria

This paper develops a novel mathematical framework for collaborative learning by means of *geometrically inspired kernel machines* which includes statements on the bounds of generalisation and approximation errors, and sample complexity. For classification problems, this approach allows us to learn bounded geometric structures around given data points and hence solve the global model learning problem in an efficient way by exploiting convexity properties of the related optimisation problem in a Reproducing Kernel Hilbert Space (RKHS). In this way, we can reduce classification problems to determining the closest *bounded geometric structure* from a given data point. Further advantages that come with our solution is that our approach does not require clients to perform multiple epochs of local optimisation using stochastic gradient descent, nor require rounds of communication between client/server for optimising the global model. We highlight that numerous experiments have shown that the proposed method is a competitive alternative to the state-of-the-art.

JAIR Associate Editor: Alessandro Farinelli

JAIR Reference Format:
Mohit Kumar, Alexander Valentinitsch, Magdalena Fuchs, Mathias Brucker, Juliana Bowles, Adnan Husakovic, Ali Abbas, and Bernhard A. Moser. 2025. Geometrically Inspired Kernel Machines for Collaborative Learning Beyond Gradient Descent. *Journal of Artificial Intelligence Research* 83, Article 16 (July 2025), 35 pages. DOI: 10.1613/jair.1.16821

*Corresponding Author.

Authors' Contact Information: Mohit Kumar, ORCID: https://orcid.org/0000-0002-7368-5157, mohit.kumar@uni-rostock.de, University of Rostock, Rostock, Mecklenburg-Vorpommern, Germany and Software Competence Center Hagenberg GmbH, Hagenberg, Upper Austria, Austria; Alexander Valentinitsch, ORCID: https://orcid.org/0000-0003-3348-6302, alexander.valentinitsch@scch.at, Software Competence Center Hagenberg GmbH, Hagenberg, Upper Austria, Austria; Magdalena Fuchs, ORCID: https://orcid.org/0009-0003-5621-1560, magdalena.fuchs@mtec.ethz.ch, ETH Zürich, Zürich, Switzerland; Mathias Brucker, ORCID: https://orcid.org/0009-0007-2670-8778, mathias.brucker@scch.at, Software Competence Center Hagenberg GmbH, Hagenberg, Upper Austria, Austria; Juliana Bowles, ORCID: https://orcid.org/0000-0002-5918-9114, jkfb@st-andrews.ac.uk, University of St Andrews, St Andrews, UK; Adnan Husakovic, ORCID: https://orcid.org/0000-0001-7150-8782, adnan.husakovic@primetals.com, Primetals Technologies Austria GmbH, Linz, Upper Austria, Austria; Ali Abbas, ORCID: https://orcid.org/0000-0001-7226-9293, ali.abbas@primetals.com, Primetals Technologies Austria GmbH, Linz, Upper Austria, Austria; Bernhard A. Moser, ORCID: https://orcid.org/0000-0001-8373-7523, bernhard.moser@scch.at, Johannes Kepler University, Linz, Upper Austria, Austria and Software Competence Center Hagenberg GmbH, Hagenberg, Upper Austria, Austria.

DOI: 10.1613/jair.1.16821

1 Introduction

In different domains of modern industry such as iron- or steelmaking, substantial amounts of data are generated and fused together to describe complex processes. Such data is often distributed originating from multiple data vendors and clients, making collaborative data analysis and model training challenging. A potential bottleneck of using traditional centralized machine learning approaches is the overall data aggregation to a single location. The centralization of the data is not always feasible due to privacy concerns and logistical constraints.

Our focus here is on collaborative learning from distributed and privately owned data. Federated learning is an increasingly popular approach to collaborative learning between multiple clients without the need to exchange raw training data. Given this advantage, federated learning can play a crucial role in process industry by leveraging distributed data to improve model performance while preserving data privacy. The classical federated learning approach [38, 33, 48] aims to train a common global model by repeating the following two operations: 1) training client local models using local data, and 2) aggregating these local models to update a global model. However, the sampling of distributed data from different local distributions makes it challenging to design and analyse efficient federated learning algorithms in practice [50, 46]. In general, essential requirements for federated learning include: 1) the capability of addressing heterogeneity among local data distributions, 2) the support for communication efficiency (allowing clients to transfer the required amount of parameters to the server under limited communication bandwidth), and 3) overall computational efficiency (for real-time operations) [15].

1.1 Central Problem

In order to develop an accurate collaborative learning method that is efficient in both communication and computation, we formulate the following research questions:

> *Q1: Can we build a theoretical analysis framework for collaborative learning from distributed and statistically heterogeneous data that, without making any assumptions on data distributions, allows us to calculate in practice: 1) the mean squared generalisation and approximation error bounds, and 2) the minimum number of training samples required to reduce the risk in approximating the target function below ϵ (for any $\epsilon > 0$) with probability at least $1 - \delta$ (for any $\delta \in (0, 1)$)?*
> *Q2: Can we solve the global model optimisation problem in the federated setting without requiring multiple rounds of communication between clients and the server?*
> *Q3: Can kernel machine learning theory provide a competitive and computationally efficient alternative to stochastic gradient descent-based optimisation in a federated setting?*

1.2 The State of the Art

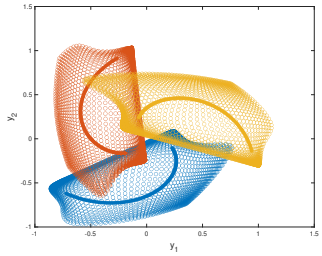
Addressing Data Heterogeneity. The issue of data heterogeneity in federated learning has been previously addressed by learning a personalised model for each client assuming that data features share a common global representation, while statistical heterogeneity across clients is attributed to the labels [2]. The personalised federated learning problem has been also studied under the model-agnostic meta-learning framework with the goal of finding an initial shared model that can easily be adapted to local datasets by performing a few steps of gradient descent [3]. Another personalised approach is that clients, instead of fully utilising the averaged global parameters for initialisation, only select a subset of the global model's parameters, and load the remaining parameters from previous local models [44]. Adversarial learning is another approach to deal with heterogeneous data features, where a discriminator is trained to distinguish the representations of the clients, while the clients aim to generate indistinguishable representations [32]. Alternatively, a clustered federated learning approach has been proposed based on the grouping of clients into clusters so that clients of the same cluster share the same model [43, 47].

Theoretical Analysis Framework. In federated learning, the underlying models can be chosen from a reproducing kernel Hilbert space [11, 4] allowing for an application of the powerful kernel theory for design and analysis. Kernels have been applied in machine learning over the years [10, 42] and have recently gained renewed attention. In particular, the parallels between the properties of deep neural networks and kernel methods have been established to indicate that some key phenomena of deep learning are manifested similarly in kernel methods in the *overfitted* regime [1], and deep kernel machines have been introduced [49, 40]. Kernel-based models are effective for learning representations [5, 16, 31] and facilitate analytical solutions for learning problems using a broad range of mathematical techniques. A convergence guarantee for federated learning can be established for strongly convex and smooth objective functions [35, 19, 41]. For one-hidden layer neural network with ReLU activations, an analysis of federated learning can be provided [36] by describing the training dynamics of federated learning by means of Neural Tangent Kernel (NTK) [13]. The gradient descent training dynamics of artificial neural networks follows that of the gradient descent of the functional cost with respect to a kernel: NTK [13]. A NTK based framework makes use of the theory on over-parameterised neural networks to provide proof of convergence of gradient descent and generalisation bound for over-parameterized ReLU neural networks in federated learning [12].

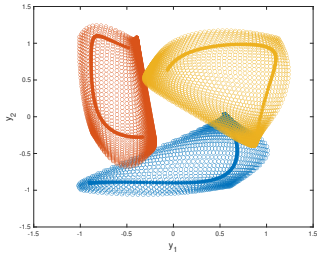
Variational Optimisation as an Alternative to Gradient Descent. A kernel-based approach that does not rely on gradient descent-based learning and instead uses variational optimisation for deriving analytically the learning solutions, has been previously studied [22, 30, 52, 28, 51, 25, 24, 29]. This kernel-based variational optimisation approach was considered for privacy-preserving learning under the differential privacy framework [21, 20, 27] and fully homomorphic encryption [29], and can potentially be explored for federated learning as well. So far there have been no attempts to extend the variational optimisation approach to federated learning in such a way that it can handle all our research questions Q1, Q2, and Q3.

Geometrically Inspired Kernel Approach as an Efficient Alternative. For collaborative learning in a federated setting, a geometrically inspired kernel approach has been introduced [26, 23]. A recent paper [26] introduced a so-called *Kernel Affine Hull Machine (KAHM)*, wherein a representation of given data points is learned in RKHS to define a bounded geometric structure around data points within the affine hull of data points. The KAHM makes it possible to compute at any arbitrary point a measure of its distance from the data samples. The significance of this is that the KAHM's induced distance measure cannot only be used for classification, but for federated learning by aggregating locally trained KAHMs to build a global KAHM. Note that the crucial significance of KAHMs for learning from distributed data comes from the fact that a *global model can be built by aggregating local models simply using a distance measure without requiring gradient-based learning of the global model parameters*. This is best illustrated through an example as shown in Fig. 1. Consequently, a KAHM-based approach is computationally more efficient and indeed promising for federated learning [23, 26]. Moreover, KAHMs can be used to mitigate the accuracy-loss issue of differential privacy, where the post-processing property of differential privacy is leveraged for fabricating new data samples by means of a geometric model ensuring that the geometric modelling error of fabricated data samples is never larger than that of original data samples while simultaneously achieving the privacy-loss bound. Although a mathematical proof of fabricated data samples with modelling error less than that of original data samples is provided [26], no generalisation error analysis and performance guarantees have been provided for the KAHM-based learning method. The federated learning solution, as suggested in [26, 23], has been introduced in a rather *adhoc* manner without providing a mathematical theory to justify the solution.

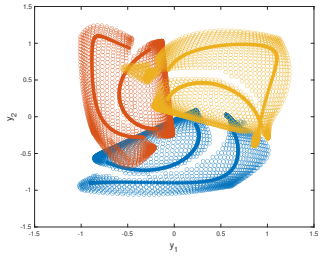
Remark 1 (Research Gap). *Existing studies on federated learning address data heterogeneity through a personalised or clustered approach, develop a theoretical analysis framework for gradient descent-based learning, and introduce a KAHM-based efficient solution without providing a mathematical theory as a suitable justification. However, these aspects have been studied separately and not together in a unified manner. Indeed, there is a lack of a*



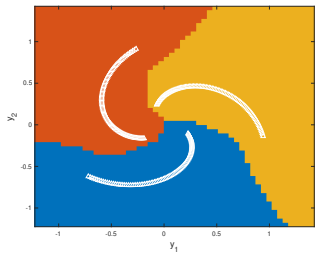
(a) Local dataset 1: A set of samples has been labelled into 3 classes and each class modelled through a KAHM. The image of each KAHM defines a bounded geometric structure around data points.



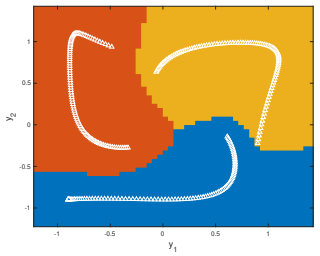
(b) Local dataset 2: Another set of samples also labelled into 3 classes with classes modelled through a KAHM. As before, the image of each KAHM defines a bounded geometric structure around data points.



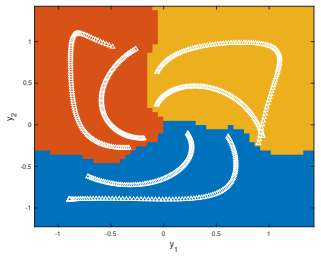
(c) Global KAHM: The local class-specific KAHMs, trained independently using dataset 1 and 2, can be aggregated to build the global class-specific KAHM using the distance measure.



(d) Decision boundary of classifier 1: the KAHM induced distance measure allows us to build a classifier from local dataset 1.



(e) Decision boundary of classifier 2: the KAHM induced distance measure also allows us to build a classifier from local dataset 2.



(f) Decision boundary of a global classifier: The global KAHMs make it possible for us to obtain a global classifier from the distributed datasets 1 and 2 in a federated setting.

Fig. 1. An illustration of the KAHM-based learning from distributed data.

unified collaborative learning framework that is powerful enough to simultaneously address our formulated research questions Q1, Q2, and Q3.

1.3 Contributions

In this study, we give an affirmative answer to Q1, Q2, and Q3 by providing a novel approach (based on KAHMs) that harnesses the distributed computational power across clients. We provide a unified framework for the design and analysis of collaborative learning by means of geometrically inspired kernel machines such as KAHMs. Our contributions are summarised in the following four aspects:

Theoretical Framework: We develop a framework to analyse KAHM-based collaborative learning in a federated setting. We introduce a *novel kernel function defined by a global KAHM (that aggregates local KAHMs) such that the kernel function evaluates the degree of similarity between two data points in terms of their distance from training data samples.* The hypothesis space for the learning is suitably (specifically, a convex hull) defined in the RKHS associated to the novel kernel function. An *upper bound on the Rademacher complexity of the hypothesis space* is provided (in Theorem 3.1) to derive a *uniform bound on the generalisation error* (in Theorem 3.3).

Beyond Gradient Descent Learning Regime: Unlike most studies, we move beyond the gradient descent learning regime to derive a collaborative learning solution, that utilises the idea of KAHM's induced distance measure based aggregation of local geometrical models to build a global geometrical model. Our contribution lies in considering the global model learning problem (in Problem 1) and showing that under a realistic assumption, it is possible to *derive analytically a learning solution* (in Theorem 4.1) that *does not require estimating global model parameters*. The underlying assumption is that there is only a small error in the fitting of training data points by the KAHM. This assumption is realistic and is validated through various experiments as well. *Since the learning solution does not require estimating the global model parameters, no rounds of communication between clients and server are required for optimising the global model, and the clients are also not required to perform multiple epochs of local optimisation using stochastic gradient descent*. The advantage of our approach is hence that it leads to a *communication and computationally efficient collaborative learning solution.*

Performance Guarantees: The generalisation error bound allows us to derive an *upper bound on the error in approximating the target function* (in Theorem 4.3). Note that the *target function approximation risk bound can be calculated in practice and decays as* $O(1/\sqrt{N})$*, where* N *is the total number of training data samples distributed across multiple clients*. Remarkably, the risk bound depends only on N, and thus the *sample complexity* (i.e. the number of training samples needed for an arbitrarily small risk in approximating the target function) is calculated (in Lemma 4.5) and plotted (in Fig. 3). We additionally provide a deterministic analysis (in Theorem 4.6) that further justifies the proposed solution via an interpretation in-terms of distance from training data points.

Competitive Alternative: A KAHM-based learning approach provides a *competitive alternative to the state of the art federated learning methods*, and in fact outperforms traditional solutions. Furthermore, the KAHM approach facilitates and enhances *cross-domain knowledge transfer in federated settings*. Experiments are used to show the improved performance of the proposed method when compared to the state of the art methods.

1.4 Proposed Approach, Novelty, and Significance

Our unified approach for the development of a collaborative learning framework for addressing our research questions (Q1, Q2, and Q3) consists of the following 9 steps:

Step 1: Define a KAHM induced kernel function. KAHMs let us define a novel kernel function that measures the similarity between two data points in terms of their distance from training samples of a class.

Step 2: Define a data-dependent hypothesis space for learning. To predict the association between a class-label and a data point, the considered hypothesis space is defined by the given data samples and is in the form of a convex hull within the RKHS associated to the KAHM induced kernel function.

Step 3: Calculate the upper bound of the Rademacher complexity of the hypothesis space. The Rademacher complexity of the considered hypothesis space has an upper bound such that this bound can be calculated in practice.

Step 4: Derive the generalisation error bound for the hypothesis space. Following the standard approach, the Rademacher complexity can be used to derive a uniform bound on the generalisation error.

Step 5: Formulate the global model learning problem. The global model learning problem can be formulated as an optimisation problem over a *suitably* chosen subset of the hypothesis space.

Step 6: Exploit the convex hull form of the hypothesis space for deriving a learning solution analytically. The convex hull form of the hypothesis space can be leveraged together with a realistic assumption to derive a

learning solution that does not require estimating any of the model parameters using gradient descent or any other numerical algorithm.

Step 7: Derive the upper bound of the error in approximating the target function. The generalization error bound can be used to derive an upper bound of the error in approximating the target function.

Step 8: Calculate the sample complexity. The target function approximation risk bound can be used to calculate the sample complexity.

Step 9. Provide a deterministic analysis of the solution. The upper bound of the KAHM induced distance function can be used to analyse and interpret the solution in terms of the distance from training data points.

Remark 2 (Novelty). *The above 9 steps offer a novel approach to the development of a collaborative learning framework. In particular, the introduction of a KAHM-induced kernel function (step 1) and exploiting the convex hull form of the hypothesis space (step 2) for deriving the learning solution analytically (step 6) are original. This is the first study applying geometrically inspired kernel machines (i.e., KAHMs) for a rigorous design and analysis of collaborative learning solutions.*

Remark 3 (Significance). *The proposed approach has been carefully designed to address the formulated research questions. Q1 is addressed by steps 4, 7, and 8. Q2 and Q3 are addressed by step 6. A new deterministic way of studying and solving the learning problem is provided by step 9. The work provides a new theoretical analysis framework for learning beyond gradient descent.*

1.5 Limitations and Future Research Directions

The KAHM approach paves the way for AutoML, yet our current work is limited by the absence of representation learning, which may necessitate feature extraction as a preprocessing step for raw data. Our future work will study representation learning by expanding a KAHM-based approach to automatically extract features from raw data (including images) at varying abstraction levels. In particular, we will focus on developing KAHM-based representations learning algorithms outside the realm of stochastic gradient descent. The other limitation of the current work is providing theoretical analysis only for the squared loss function, and thus we will also extend the results to a family of loss functions in a generalised setting. Since Rademacher complexity is a traditional technique, it is also worth investigating operator and spectral methods for geometrically inspired kernel machines. Specifically, we will study the effect of spectral properties of the kernel matrix for achieving even tighter bounds on the errors. The current study achieves robustness towards statistical heterogeneity (among clients' data) by design. Another possible extension is to establish and study the robustness of the method towards noise present in the data and quantifying the uncertainties affecting the prediction. Further, an extension of the methodology to include time series data will be considered in the future. Given that previous methods have been evaluated on benchmark datasets such as MNIST, Freiburg Groceries, CIFAR-10, CIFAR-100, and Office-Caltech-10, we confined our experiments to these datasets to enable a direct comparison with state-of-the-art techniques. In future work, we plan to evaluate our method on high-dimensional, non-image datasets.

1.6 Structure of the Paper

Section 2 presents the necessary mathematical background underlying our work. Section 3 develops the theory for KAHMs and includes steps 1-4 of the proposed approach outlined earlier. Section 4 continues with steps 5-9, thereby solving the collaborative learning problem. The experimental evaluation of our approach is given in Section 5, followed by concluding remarks in Section 6.

2 Mathematical Prerequisites and Notations

This section introduces the notation used throughout, presents the distributed data setting, and provides a review of the notion of KAHM.

2.1 Notation

In this paper, all matrices are denoted using boldface font. The following notation is used:

- Let $n, p, c, N, Q, C \in \mathbb{Z}_+$ be the positive integers.
- For a scalar $a \in \mathbb{R}$, $|a|$ denotes its absolute value. For a set A, $|A|$ denotes its cardinality. For a real matrix $\mathbf{Y}$, $\mathbf{Y}^T$ is the transpose of $\mathbf{Y}$.
- For a vector $y \in \mathbb{R}^p$, $\|y\|$ denotes the Euclidean norm. For a matrix $\mathbf{Y} \in \mathbb{R}^{N\times p}$, $\|\mathbf{Y}\|_2$ denotes the spectral norm and $\|\mathbf{Y}\|_F$ denotes the Frobenius norm.
- For a vector $y \in \mathbb{R}^p$, y_j (and also $(y)_j$) denotes the j-th element. For a matrix $\mathbf{Y} \in \mathbb{R}^{N\times p}$, $(\mathbf{Y})_{i,:}$ denotes the i-th row, $(\mathbf{Y})_{:,j}$ denotes the j-th column, and $(\mathbf{Y})_{i,j}$ denotes the (i, j)-th element.
- For a set $\{y^1, \cdots, y^N\} \subset \mathbb{R}^p$, its affine hull is denoted as $\mathrm{aff}(\{y^1, \cdots, y^N\})$.
- The square brackets are used to represent the construction of a matrix from columns e.g. $\begin{bmatrix} y^1 & \cdots & y^N \end{bmatrix}$ is a matrix with vectors $y^1, \cdots, y^N$ as the columns.
- Let $\mathcal{X} \subset \mathbb{R}^n$ be a region. A RKHS, $\mathcal{H}_k(\mathcal{X})$, is a Hilbert space of functions $f : \mathcal{X} \to \mathbb{R}$ on a non-empty set $\mathcal{X}$ with a reproducing kernel $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ satisfying $\forall x \in \mathcal{X}$ and $\forall f \in \mathcal{H}$,

$$k(\cdot, x) \in \mathcal{H}_k(\mathcal{X}), \tag{1}$$

$$\langle f, k(\cdot, x)\rangle_{\mathcal{H}_k(\mathcal{X})} = f(x), \tag{2}$$

  where $\langle\cdot,\cdot\rangle_{\mathcal{H}_k(\mathcal{X})} : \mathcal{H}_k(\mathcal{X}) \times \mathcal{H}_k(\mathcal{X}) \to \mathbb{R}$ is an inner product on $\mathcal{H}_k(\mathcal{X})$. Let $\|f\|_{\mathcal{H}_k(\mathcal{X})} \coloneqq \sqrt{\langle f, f\rangle_{\mathcal{H}_k(\mathcal{X})}}$ denote the norm induced by the inner product on $\mathcal{H}_k(\mathcal{X})$.
- Let $\mathbf{K}$ be a symmetric matrix, $\mathbf{K} \succ 0$ denotes that $\mathbf{K}$ is positive-definite.
- Let $(\mathcal{S}_{y,z,q}, \mathcal{F}_{y,z,q}, \mu_{y,z,q})$ be a *probability space* and $(y, z, q) : \mathcal{S}_{y,z,q} \to \mathbb{R}^p \times \{0, 1\}^C \times \{1, 2, \cdots, Q\}$ be a random vector on $\mathcal{S}_{y,z,q}$. Let $\mathcal{B}(\mathbb{R}^p \times \{0, 1\}^C \times \{1, 2, \cdots, Q\})$ denote the Borel $\sigma-$algebra on $\mathbb{R}^p \times \{0, 1\}^C \times \{1, 2, \cdots, Q\}$. Let $\mathbb{P}_{y,z,q} : \mathcal{B}(\mathbb{R}^p \times \{0, 1\}^C \times \{1, 2, \cdots, Q\}) \to \mathbb{R}$ be the distribution of (y, z, q) given as

$$\mathbb{P}_{y,z,q} \coloneqq \mu_{y,z,q} \circ (y, z, q)^{-1}. \tag{3}$$

2.2 Distributed Data Setting

We consider the multi-class problem, where a label vector $z \in \{0, 1\}^C$ with $C \geq 2$ is assigned to a data point $y \in \mathbb{R}^p$ such that the c-th element of vector z, $z_c \in \{0, 1\}$, represents the association of y with the c-th class. We consider the problem of collaborative learning from distributed data, where a number of clients Q ($Q \geq 1$) participate in the learning. Let $q \in \{1, 2, \cdots, Q\}$ be the client characterising variable. Let $\mathcal{D}$ be a set consisting of N number of samples drawn i.i.d. according to the distribution $\mathbb{P}_{y,z,q}$:

$$\mathcal{D} \coloneqq \left\{(y^i, z^i, q^i) \in \mathbb{R}^p \times \{0, 1\}^C \times \{1, 2, \cdots, Q\} \;\middle|\; i \in \{1, 2, \cdots, N\}\right\} \sim (\mathbb{P}_{y,z,q})^N. \tag{4}$$

Let $\mathcal{I}^{c,q}$ be the set of indices of those samples in the sequence $\left((y^i, z^i, q^i) \in \mathcal{D}\right)_{i=1}^N$ which are c-th class labelled and owned by client q, i.e.,

$$\mathcal{I}^{c,q} \coloneqq \left\{i \in \{1, 2, \cdots, N\} \;\middle|\; (z^i)_c = 1,\ q^i = q\right\}. \tag{5}$$

Let $(\mathrm{I}_1^{c,q}, \cdots, \mathrm{I}_{|\mathcal{I}^{c,q}|}^{c,q})$ be the sequence of elements of $\mathcal{I}^{c,q}$ in ascending order, i.e.,

$$\mathrm{I}_1^{c,q} = \min(\mathcal{I}^{c,q}), \tag{6}$$

$$\mathrm{I}_i^{c,q} = \min(\mathcal{I}^{c,q} \setminus \{\mathrm{I}_1^{c,q}, \cdots, \mathrm{I}_{i-1}^{c,q}\}), \tag{7}$$

for $i \in \{2, \cdots, |\mathcal{I}^{c,q}|\}$. Let $\mathbf{Y}^{c,q} \in \mathbb{R}^{|\mathcal{I}^{c,q}| \times p}$ be the matrix storing the c-th class labelled and q-th client owned samples, i.e.,

$$\mathbf{Y}^{c,q} = \begin{bmatrix} y^{\mathrm{I}_1^{c,q}} & \cdots & y^{\mathrm{I}_{|\mathcal{I}^{c,q}|}^{c,q}} \end{bmatrix}^T. \tag{8}$$

Remark 4 (Addressing Statistical Heterogeneity). *Our analysis takes into account the heterogeneity among client's data distributions by automatically considering, for arbitrary clients q^i and q^j with $i \neq j$, the following cases as well:*

$$\mathbb{P}_{y,z|q}(\cdot, \cdot|q = q^i) \neq \mathbb{P}_{y,z|q}(\cdot, \cdot|q = q^j), \tag{9}$$

$$\mathbb{P}_{z|y,q}(\cdot|y, q = q^i) \neq \mathbb{P}_{z|y,q}(\cdot|y, q = q^j). \tag{10}$$

Hence, data shifts among clients are being considered.

2.3 A Review of Kernel Affine Hull Machines

We recall the notion of KAHM as originally defined in [26].

Definition 2.1 (Kernel Affine Hull Machine (KAHM) [26]). Given a finite number of samples: $\mathbf{Y} = \begin{bmatrix} y^1 & \cdots & y^N \end{bmatrix}^T$ with $y^1, \cdots, y^N \in \mathbb{R}^p$ and a subspace dimension $n \leq p$; a kernel affine hull machine $\mathcal{A}_{\mathbf{Y},n} : \mathbb{R}^p \longrightarrow \mathrm{aff}(\{y^1, \cdots, y^N\})$ maps an arbitrary point $y \in \mathbb{R}^p$ onto the affine hull of $\{y^1, \cdots, y^N\}$.

The complete definition of $\mathcal{A}_{\mathbf{Y},n}$ is provided in Appendix A. In addition, the subspace dimension $n \leq p$ is practically determined for the given samples using a procedure given in Appendix B.

Remark 5 (KAHM for Automated Machine Learning (AutoML)). *A KAHM $\mathcal{A}_{\mathbf{Y},n}$, with the choice of subspace dimension n as suggested in Appendix B, is completely defined by the data samples Y without involving any free parameters to be tuned, allowing us to define an AutoML approach by means of KAHMs. Consequently, in what follows we use $\mathcal{A}_{\mathbf{Y}}$ to denote a KAHM.*

Theorem 2.2 (KAHM as a Bounded Function [26]). *The KAHM $\mathcal{A}_{\mathbf{Y}}$, associated to $\mathbf{Y} = \begin{bmatrix} y^1 & \cdots & y^N \end{bmatrix}^T$ with $y^1, \cdots, y^N \in \mathbb{R}^p$, is a bounded function on $\mathbb{R}^p$ such that for any $y \in \mathbb{R}^p$,*

$$\|\mathcal{A}_{\mathbf{Y}}(y)\| < \|\mathbf{Y}\|_2 \left(1 + \frac{pN^2}{2\|\mathbf{Y}\|_F^2}\right). \tag{11}$$

Thus, the image of $\mathcal{A}_{\mathbf{Y}}$ is bounded such that

$$\mathcal{A}_{\mathbf{Y}}[\mathbb{R}^p] \subset \left\{ y \in \mathbb{R}^p \;\middle|\; \|y\| < \|\mathbf{Y}\|_2 \left(1 + \frac{pN^2}{2\|\mathbf{Y}\|_F^2}\right) \right\}. \tag{12}$$

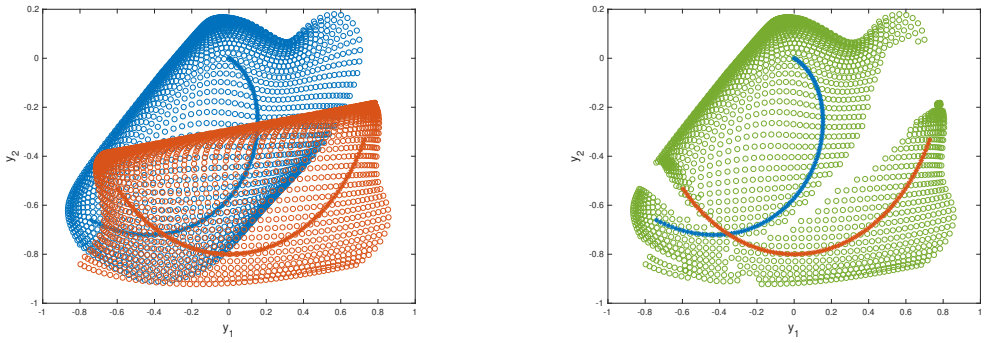
Definition 2.3 (A Distance Function Induced by KAHM [26]). Given a KAHM $\mathcal{A}_{\mathbf{Y}}$, the distance of an arbitrary point $y \in \mathbb{R}^p$ from its image under $\mathcal{A}_{\mathbf{Y}}$ is given as

$$\Gamma_{\mathcal{A}_{\mathbf{Y}}}(y) := \|y - \mathcal{A}_{\mathbf{Y}}(y)\|. \tag{13}$$

Theorem 2.4 ($\Gamma_{\mathcal{A}_{\mathbf{Y}}}(\cdot)$ as a Measure of Distance from Data Points [26]). *The ratio of the distance of a point $y \in \mathbb{R}^p$ from its image under $\mathcal{A}_{\mathbf{Y}}$ to the distance of y from $\{y^1, \cdots, y^N\}$ evaluated as $\left\| \begin{bmatrix} y - y^1 & \cdots & y - y^N \end{bmatrix} \right\|_2$ remains upper bounded as*

$$\frac{\Gamma_{\mathcal{A}_{\mathbf{Y}}}(y)}{\left\| \begin{bmatrix} y - y^1 & \cdots & y - y^N \end{bmatrix} \right\|_2} < 1 + \frac{pN^2}{2\|\mathbf{Y}\|_F^2}. \tag{14}$$

Theorem 2.4 states that if a point y is close to points $\{y^1, \cdots, y^N\}$, then the value $\Gamma_{\mathcal{A}_{\mathbf{Y}}}(y)$ cannot be large. Thus, a large value of the distance function at a point y indicates that y must be at a far distance from $\{y^1, \cdots, y^N\}$.



(a) images of two different KAHMs built independently using two different datasets

(b) image of the global KAHM combining together two independently built KAHMs

Fig. 2. An example of combining together local KAHMs to build a global KAHM.

2.4 Combining Locally Distributed KAHMs to build a Global KAHM using the Distance Function

We consider the scenario where class labelled data samples are distributed amongst Q different clients. Given Q different KAHMs $\mathcal{A}_{\mathbf{Y}^{c,1}}, \cdots, \mathcal{A}_{\mathbf{Y}^{c,q}}$ built independently using data matrices $\mathbf{Y}^{c,1}, \cdots, \mathbf{Y}^{c,q}$ respectively, a possible way to combine together the KAHMs is as follows:

$$\mathcal{G}_c(y) = \mathcal{A}_{\mathbf{Y}^{c,\hat{q}(y)}}(y) \tag{15}$$

$$\hat{q}(y) = \underset{q\in\{1,2,\cdots,Q\}}{\operatorname{argmin}} \Gamma_{\mathcal{A}_{\mathbf{Y}^{c,q}}}(y), \tag{16}$$

where $\mathcal{G}_c$ is the global KAHM (that combines together the individual KAHMs) and $\Gamma_{\mathcal{A}_{\mathbf{Y}^{c,q}}}$ is the distance function induced by $\mathcal{A}_{\mathbf{Y}^{c,q}}$. A 2-dimensional data example where two different KAHMs are combined to build a global KAHM is provided in Figure 2. Figure 2 shows the images of individual KAHMs (in Figure 2(a)) and the image of global KAHM (in Figure 2(b)).

Definition 2.5 (A Distance Function Induced by Global KAHM). Given a global KAHM ($\mathcal{G}_c$) that combines together Q local KAHMs ($\mathcal{A}_{\mathbf{Y}^{c,1}}, \cdots, \mathcal{A}_{\mathbf{Y}^{c,q}}$), the distance of an arbitrary point $y \in \mathbb{R}^p$ from its image under $\mathcal{G}_c$ is given as

$$\Gamma_{\mathcal{G}_c}(y) := \|y - \mathcal{G}_c(y)\| \tag{17}$$

$$= \min_{q\in\{1,2,\cdots,Q\}} \Gamma_{\mathcal{A}_{\mathbf{Y}^{c,q}}}(y). \tag{18}$$

3 Theory for Learning with Kernel Affine Hull Machines

This section develops a theory for learning by means of KAHMs. For this, a geometrically inspired kernel function and corresponding RKHS is presented in Section 3.1, followed by a definition of a hypothesis space in Section 3.2. Section 3.3 evaluates the Rademacher complexity of the hypothesis space, which is then used in Section 3.4 to derive the generalisation error bound for the hypothesis space.

3.1 A Novel Kernel Function Induced by the Global KAHM

A given global KAHM, $\mathcal{G}_c$, induces a function, $\mathcal{K}_c : \mathbb{R}^p \times \mathbb{R}^p \to [0,1]$, defined as

$$\mathcal{K}_c(y^1, y^2) := \exp\left(-\frac{1}{p}\Gamma_{\mathcal{G}_c}(y^1)\right)\exp\left(-\frac{1}{p}\Gamma_{\mathcal{G}_c}(y^2)\right). \tag{19}$$

$\mathcal{K}_c$ is a positive definite kernel, since

(1) $\mathcal{K}_c(y^1, y^2) = \mathcal{K}_c(y^2, y^1)$, and
(2) for every $y^1, \cdots, y^N \in \mathbb{R}^p$ and $\alpha_1, \cdots, \alpha_N \in \mathbb{R}$,

$$\sum_{i,j=1}^{N} \alpha_i \alpha_j \mathcal{K}_c(y^i, y^j) \geq 0. \tag{20}$$

To see (20), consider

$$\begin{aligned}\sum_{i,j=1}^{N} \alpha_i \alpha_j \mathcal{K}_c(y^i, y^j) &= \sum_{i,j=1}^{N} \alpha_i \exp\left(-\frac{1}{p}\Gamma_{\mathcal{G}_c}(y^i)\right)\alpha_j \exp\left(-\frac{1}{p}\Gamma_{\mathcal{G}_c}(y^j)\right)\\ &= \left|\sum_{i=1}^{N} \alpha_i \exp\left(-\frac{1}{p}\Gamma_{\mathcal{G}_c}(y^i)\right)\right|^2 \geq 0.\end{aligned} \tag{21}$$

Remark 6 (Justification and Interpretation of $\mathcal{K}_c$). *$\mathcal{K}_c(y^1, y^2)$ will be high, only if both y^1 and y^2 lie close to the c-th class labelled samples that may have been owned by any of the Q clients. Thus, $\mathcal{K}_c$ provides a measure of similarity between two data points in-terms of their association to the c-th class. That is, $\mathcal{K}_c(y^1, y^2)$ will be high even in the case when y^1 and y^2 are at far distance from each other but both y^1 and y^2 are close to some c-th class labelled samples. This property of $\mathcal{K}_c$ justifies its choice as kernel function for predicting the association of a point to c-th class.*

Now the RKHS associated to $\mathcal{K}_c$ is given by:

$$\begin{aligned}&\mathcal{H}_{\mathcal{K}_c}(\mathbb{R}^p)\\ &:= \left\{ f = \sum_{i=1}^{\infty} \alpha_i \mathcal{K}_c(\cdot, y^i) \;\mid\; \alpha_i \in \mathbb{R},\ y^i \in \mathbb{R}^p,\ \|f\|^2_{\mathcal{H}_{\mathcal{K}_c}(\mathbb{R}^p)} := \sum_{i,j=1}^{\infty} \alpha_i \alpha_j \mathcal{K}_c(y^i, y^j) < \infty \right\}\end{aligned} \tag{22}$$

with inner product for any $f(\cdot) = \sum_{i=1}^{L} a_i \mathcal{K}_c(\cdot, s^i)$ (with $a_i \in \mathbb{R}, s^i \in \mathbb{R}^p$) and $g(\cdot) = \sum_{j=1}^{M} b_j \mathcal{K}_c(\cdot, t^j) \in \mathcal{H}_{\mathcal{K}_c}(\mathbb{R}^p)$ (with $b_j \in \mathbb{R}, t^j \in \mathbb{R}^p$) defined as

$$\langle f, g \rangle_{\mathcal{H}_{\mathcal{K}_c}(\mathbb{R}^p)} := \sum_{i=1}^{L}\sum_{j=1}^{M} a_i b_j \mathcal{K}_c(s^i, t^j). \tag{23}$$

3.2 A Data-Dependent Hypothesis Space

Let $f_{y \mapsto z_c} : \mathbb{R}^p \to \mathbb{R}$ be a function such that $f_{y \mapsto z_c}(y)$ serves as an approximation to z_c, i.e., $f_{y \mapsto z_c}(y)$ predicts the association of the data point y to the c-th class. Given the data set $\mathcal{D}$, as defined in (4), the hypothesis space for predicting the association of a point to the c-th class is defined as a convex hull within $\mathcal{H}_{\mathcal{K}_c}(\mathbb{R}^p)$:

$$\mathcal{M}_{\mathcal{D},c} := \left\{ f_{y \mapsto z_c} = \sum_{i=1}^{N} \alpha_{c,i} \mathcal{K}_c(\cdot, y^i) \;\mid\; \alpha_{c,i} \in [0,1],\ \sum_{i=1}^{N} \alpha_{c,i} = 1,\ (y^i, z^i, q^i) \in \mathcal{D} \right\}. \tag{24}$$

It is obvious that

$$\mathcal{M}_{\mathcal{D},c} \subset \mathcal{H}_{\mathcal{K}_c}(\mathbb{R}^p). \tag{25}$$

It can be seen that for any $f_{y\mapsto z_c} \in \mathcal{M}_{\mathcal{D},c}$,

$$\left\|f_{y\mapsto z_c}\right\|_{\mathcal{H}_{\mathcal{K}_c}(\mathbb{R}^p)} = \left|\sum_{i=1}^{N} \alpha_{c,i} \exp\left(-\frac{1}{p}\Gamma_{\mathcal{G}_c}(y^i)\right)\right| \tag{26}$$

$$\leq 1, \tag{27}$$

where (27) follows from $\alpha_{c,i} \in [0,1]$ and $\sum_{i=1}^{N} \alpha_{c,i} = 1$. Thus,

$$\sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}} \left\|f_{y\mapsto z_c}\right\|_{\mathcal{H}_{\mathcal{K}_c}(\mathbb{R}^p)} \leq 1. \tag{28}$$

3.3 Rademacher Complexity of the Hypothesis Space

To evaluate the Rademacher complexity of the hypothesis space, we introduce $\sigma_1, \cdots, \sigma_N$ as the independent random variables drawn from the Rademacher distribution, and denote $\sigma = (\sigma_1, \cdots, \sigma_N)$. For a given data set $\mathcal{D}$ (defined in (4)), the empirical Rademacher complexity of the hypothesis space $\mathcal{M}_{\mathcal{D},c}$ is given as

$$\widehat{\mathcal{R}}_{\mathcal{D}}(\mathcal{M}_{\mathcal{D},c}) = \frac{1}{N}\mathop{\mathbb{E}}_{\sigma}\left[\sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}} \sum_{i=1}^{N} \sigma_i\, f_{y\mapsto z_c}(y^i)\right]. \tag{29}$$

Theorem 3.1 (Bound on the Rademacher Complexity of $\mathcal{M}_{\mathcal{D},c}$). *Given a dataset $\mathcal{D} \sim (\mathbb{P}_{y,z,q})^N$, as defined in (4), we have*

$$\widehat{\mathcal{R}}_{\mathcal{D}}(\mathcal{M}_{\mathcal{D},c}) \leq \frac{1}{\sqrt{N}}. \tag{30}$$

Thus, the expected Rademacher complexity has an upper bound given by

$$\mathop{\mathbb{E}}_{\mathcal{D}\sim(\mathbb{P}_{y,z,q})^N}\left[\widehat{\mathcal{R}}_{\mathcal{D}}(\mathcal{M}_{\mathcal{D},c})\right] \leq \frac{1}{\sqrt{N}}. \tag{31}$$

Proof. The proof is provided in Appendix C. □

3.4 Generalization Error Bound

We consider the squared loss function and derive generalisation error bound for our hypothesis space. Given a hypothesis $f_{y\mapsto z_c} \in \mathcal{M}_{\mathcal{D},c}$, let $l_{f_{y\mapsto z_c}} : \mathbb{R}^p \times \{0,1\} \to \mathbb{R}$ be a loss function defined as

$$l_{f_{y\mapsto z_c}}(y, z_c) := |z_c - f_{y\mapsto z_c}(y)|^2. \tag{32}$$

Consider the following family of loss functions defined by the hypothesis space $\mathcal{M}_{\mathcal{D},c}$:

$$\mathcal{L}_{\mathcal{D},c} := \left\{l_{f_{y\mapsto z_c}} : (y, z_c) \mapsto |z_c - f_{y\mapsto z_c}(y)|^2 \;\middle|\; f_{y\mapsto z_c} \in \mathcal{M}_{\mathcal{D},c}\right\}. \tag{33}$$

The empirical Rademacher complexity of $\mathcal{L}_{\mathcal{D},c}$ is given as

$$\widehat{\mathcal{R}}_{\mathcal{D}}(\mathcal{L}_{\mathcal{D},c}) = \frac{1}{N}\mathop{\mathbb{E}}_{\sigma}\left[\sup_{l_{f_{y\mapsto z_c}}\in\mathcal{L}_{\mathcal{D},c}} \sum_{i=1}^{N} \sigma_i l_{f_{y\mapsto z_c}}(y^i, (z^i)_c)\right]. \tag{34}$$

Lemma 3.2. *Given a dataset $\mathcal{D} \sim (\mathbb{P}_{y,z,q})^N$, as defined in (4), we have*

$$\widehat{\mathcal{R}}_{\mathcal{D}}(\mathcal{L}_{\mathcal{D},c}) \leq \frac{2}{\sqrt{N}}, \tag{35}$$

$$\mathop{\mathbb{E}}_{\mathcal{D}\sim(\mathbb{P}_{y,z,q})^N}\left[\widehat{\mathcal{R}}_{\mathcal{D}}(\mathcal{L}_{\mathcal{D},c})\right] \leq \frac{2}{\sqrt{N}}. \tag{36}$$

Proof. The proof is based on the application of Talagrand's lemma [39] to get $\widehat{\mathcal{R}}_{\mathcal{D}}(\mathcal{L}_{\mathcal{D},c}) \leq 2\widehat{\mathcal{R}}_{\mathcal{D}}(\mathcal{M}_{\mathcal{D},c})$. Now, (35) and (36) follow immediately from Theorem 3.1. For the sake of completeness, a proof starting from scratch is provided in Appendix D. □

Theorem 3.3 (Data-Dependent Bound on Generalisation Error for Hypothesis Space). *Given a dataset $\mathcal{D} \sim (\mathbb{P}_{y,z,q})^N$ (as defined in (4)), for any hypothesis $f_{y\mapsto z_c} \in \mathcal{M}_{\mathcal{D},c}$, the following holds with probability at least $1-\delta$ for any $\delta \in (0,1)$:*

$$\mathop{\mathbb{E}}_{(y,z)\sim\mathbb{P}_{y,z}}[l_{f_{y\mapsto z_c}}(y,z_c)] \leq \widehat{\mathbb{E}}_{\mathcal{D}}[l_{f_{y\mapsto z_c}}] + \frac{4}{\sqrt{N}} + \sqrt{\frac{\log(1/\delta)}{2N}}, \tag{37}$$

where $l_{f_{y\mapsto z_c}}$ is the loss function (32) and $\widehat{\mathbb{E}}_{\mathcal{D}}[l_{f_{y\mapsto z_c}}]$ is the empirical averaged loss value given as

$$\widehat{\mathbb{E}}_{\mathcal{D}}[l_{f_{y\mapsto z_c}}] = \frac{1}{N}\sum_{i=1}^{N} l_{f_{y\mapsto z_c}}(y^i,(z^i)_c). \tag{38}$$

Proof. The proof is provided in Appendix E. □

Remark 7 (Choice of the Loss Function). *We have considered the squared loss function for our analysis, nevertheless, the analysis can be extended to any $\rho-$Lipschitz loss function.*

4 KAHMs Based Collaborative Learning

This section shows how the global model learning problem can be formulated mathematically (cf. Section 4.1) and solved (cf. Section 4.2) to derive a predictor. In addition, theoretical guarantees on the performance of the proposed predictor are provided in Section 4.3. The obtained theoretical results are applied in Section 4.4 to solve the classification problem in a federated setting.

4.1 Global Model Learning Problem

The model learning problem consists of determining a function $f_{y\mapsto z_c} \in \mathcal{M}_{\mathcal{D},c}$, such that the generalisation error over that function, written $\mathbb{E}_{(y,z)\sim\mathbb{P}_{y,z}}[l_{f_{y\mapsto z_c}}(y,(z)_c)]$, is as small as possible. Since $f_{y\mapsto z_c} \in \mathcal{M}_{\mathcal{D},c}$, we have $f_{y\mapsto z_c}(y) = \sum_{i=1}^{N}\alpha_{c,i}\mathcal{K}_c(y,y^i)$, with $\alpha_{c,i} \in [0,1]$ and $\sum_{i=1}^{N}\alpha_{c,i} = 1$. The value $\mathcal{K}_c(y,y^i)$ will be close to 1 if and only if $\Gamma_{\mathcal{G}_c}(y) \approx 0$ and $\Gamma_{\mathcal{G}_c}(y^i) \approx 0$ (i.e. if and only if both y and y^i lie close to the c-th class labelled training data samples). Similarly $\mathcal{K}_c(y,y^i)$ will be close to 0 if either or both y and y^i lie far away from the c-th class labelled training data samples. Based on the observation that the value $\mathcal{K}_c(y,y^i)$ is the degree of similarity between y and y^i in terms of their distance from the c-th class labelled training data samples, $\alpha_{c,i}$ (which is the weight assigned to $\mathcal{K}_c(y,y^i)$ in estimating z_c) can be chosen as

$$\alpha_{c,i} = 0,\ i \notin \bigcup_{q=1}^{Q} \mathcal{I}^{c,q}, \tag{39}$$

where $\mathcal{I}^{c,q}$ (as defined in (5)) is the set of indices of the samples which are c-th class labelled and owned by q-th party. Eq. (39) implies that the *weight assigned to a non c-th class labelled training data sample in estimating z_c is zero*. As a result of (39), our learning space (within the hypothesis space $\mathcal{M}_{\mathcal{D},c}$) is given as

$$\widetilde{\mathcal{M}}_{\mathcal{D},c}$$

$$:= \left\{ f_{y\mapsto z_c} = \sum_{q=1}^{Q}\sum_{i=1}^{\Gamma^{c,q}_{|\mathcal{I}^{c,q}|}} \alpha_{c,i}\mathcal{K}_c(\cdot,y^i) \ \mid\ \alpha_{c,i}\in[0,1],\ \sum_{q=1}^{Q}\sum_{i=1}^{\Gamma^{c,q}_{|\mathcal{I}^{c,q}|}}\alpha_{c,i} = 1,\ (y^i,z^i,q^i)\in\mathcal{D} \right\} \tag{40}$$

$$\subset \mathcal{M}_{\mathcal{D},c}. \tag{41}$$

With $\widetilde{\mathcal{M}}_{\mathcal{D},c}$ as the learning space, the learning problem can be formulated as

Problem 1 (Learning Problem). *Given a dataset $\mathcal{D} \sim (\mathbb{P}_{y,z,q})^N$ (as defined in (4)), the learning problem is formulated as*

$$f^*_{y\mapsto z_c} = \underset{f_{y\mapsto z_c}\in\widetilde{\mathcal{M}}_{\mathcal{D},c}}{\operatorname{argmin}} \underset{(y,z)\sim\mathbb{P}_{y,z}}{\mathbb{E}} \left[\left|z_c - f_{y\mapsto z_c}(y)\right|^2\right]. \tag{42}$$

4.2 A Solution of the Learning Problem

The solution of Problem 1 is challenging without making any assumptions about the unknown distribution $\mathbb{P}_{y,z,q}$. However, fortunately a workaround exists for approximating $f^*_{y\mapsto z_c}$ without directly solving Problem 1. For this, a *realistic assumption* is made:

Assumption 1. *The training data samples of c-th class (i.e. $\{y^i \mid i \in \cup_{q=1}^{Q} \mathcal{I}^{c,q}\}$) are fitted by the global KAHM $\mathcal{G}_c$ with sufficient accuracy such that*

$$\exp\left(-\frac{\Gamma_{\mathcal{G}_c}(y^i)}{p}\right) \approx 1, \ \forall i \in \bigcup_{q=1}^{Q} \mathcal{I}^{c,q}. \tag{43}$$

Assumption 1 requires that $y^i \approx \mathcal{G}_c(y^i)$, i.e. $\Gamma_{\mathcal{G}_c}(y^i) \approx 0, \forall i \in \cup_{q=1}^{Q} \mathcal{I}^{c,q}$. In other words, the fitting error on the training data samples by KAHM should be small, which is a realistic assumption. This assumption will be also validated through experiments later on.

Theorem 4.1 (An Approximate Solution to the Learning Problem). *Given a dataset $\mathcal{D} \sim (\mathbb{P}_{y,z,q})^N$ (as defined in (4)), the solution of Problem 1, under Assumption 1, is given by*

$$f^*_{y\mapsto z_c}(y) \approx \exp\left(-\frac{\Gamma_{\mathcal{G}_c}(y)}{p}\right). \tag{44}$$

Proof. Problem 1 can be formulated as

$$f^*_{y\mapsto z_c} = \sum_{q=1}^{Q} \sum_{i=\mathrm{I}_1^{c,q}}^{\mathrm{I}^{c,q}_{|\mathcal{I}^{c,q}|}} \alpha^*_{c,i} \mathcal{K}_c(\cdot, y^i), \ \text{where} \tag{45}$$

$$\begin{aligned}
&\left\{\alpha^*_{c,i} \mid i \in \bigcup_{q=1}^{Q} \mathcal{I}^{c,q}\right\} \\
&= \underset{\left\{\alpha_{c,i} \mid \alpha_{c,i}\in[0,1],\ \sum_{q=1}^{Q}\sum_{i=\mathrm{I}_1^{c,q}}^{\mathrm{I}^{c,q}_{|\mathcal{I}^{c,q}|}} \alpha_{c,i} = 1\right\}}{\operatorname{argmin}} \underset{(y,z)\sim\mathbb{P}_{y,z}}{\mathbb{E}} \left[\left|z_c - \sum_{q=1}^{Q} \sum_{i=\mathrm{I}_1^{c,q}}^{\mathrm{I}^{c,q}_{|\mathcal{I}^{c,q}|}} \alpha_{c,i} \mathcal{K}_c(y, y^i)\right|^2\right]
\end{aligned} \tag{46}$$

As a result of Assumption 1,

$$f^*_{y\mapsto z_c}(y) \approx \exp\left(-\frac{\Gamma_{\mathcal{G}_c}(y)}{p}\right) \sum_{q=1}^{Q} \sum_{i=\mathrm{I}_1^{c,q}}^{\mathrm{I}^{c,q}_{|\mathcal{I}^{c,q}|}} \alpha^*_{c,i}. \tag{47}$$

Since

$$\sum_{q=1}^{Q} \sum_{i=\mathrm{I}_1^{c,q}}^{\mathrm{I}_{|\mathcal{I}^{c,q}|}^{c,q}} \alpha_{c,i}^{*} = 1, \tag{48}$$

we get (44). □

Remark 8 (Significance of Theorem 4.1). *Theorem 4.1 is a key contribution of this paper, offering an analytical learning solution that can be computed efficiently without requiring any additional algorithms. The significance of Theorem 4.1's approximate solution lies in the fact that $f^{*}_{y\mapsto z_c}(y)$ can be evaluated without estimating any of the model parameters, leading to an efficient solution that does not require a gradient-based learning algorithm. Assumption 1 enabled us to derive an efficient approximate solution for Problem 1, a task that would otherwise have been difficult due to the unknown data distributions.*

4.3 Theoretical Guarantees

Since $f^{*}_{y\mapsto z_c} \in \mathcal{M}_{\mathcal{D},c}$, an upper bound on the generalisation error of $f^{*}_{y\mapsto z_c}$ is provided by Theorem 3.3. However, a more tight bound can be obtained in the light of Assumption 1 and by making another reasonable assumption:

Assumption 2. *The non c-th class training data samples (that is, $\{y^i \mid i \notin \cup_{q=1}^{Q} \mathcal{I}^{c,q}\}$) are not well fitted by the c-th class associated global KAHM $\mathcal{G}_c$ such that*

$$\exp\left(-\frac{\Gamma_{\mathcal{G}_c}(y^i)}{p}\right) \approx 0, \ \forall i \notin \bigcup_{q=1}^{Q} \mathcal{I}^{c,q}. \tag{49}$$

Assumption 2 is reasonable, since class c's associated global KAHM $\mathcal{G}_c$ has been learned to fit only the class c labelled samples, and thus non-c class labelled training samples can not be reconstructed using $\mathcal{G}_c$, resulting in a large value of $\Gamma_{\mathcal{G}_c}$ at non-c class labelled training samples. We will evaluate the validity of Assumption 2 through experiments.

Theorem 4.2 (Bound on Generalization Error of Predictor $f^{*}_{y\mapsto z_c}$). *Given a dataset $\mathcal{D} \sim (\mathbb{P}_{y,z,q})^N$ (as defined in (4)), for the predictor $f^{*}_{y\mapsto z_c}$ (as defined in (44)), under Assumption 1 and Assumption 2, the following holds with probability at least $1-\delta$ for any $\delta \in (0,1)$:*

$$\mathop{\mathbb{E}}_{(y,z)\sim\mathbb{P}_{y,z}} \left[\left|z_c - f^{*}_{y\mapsto z_c}(y)\right|^2\right] \leq \frac{4}{\sqrt{N}} + \sqrt{\frac{\log(1/\delta)}{2N}}. \tag{50}$$

Proof. The proof is provided in Appendix F. □

Since we are considering the multi-class classification problem, an analysis of the error in approximating the class label probability (conditioned on a given input sample) is of interest. We are interested in upper bounding the mismatch between predictor $f^{*}_{y\mapsto z_c}(y)$ and the target function $\mathbb{P}_{z|y}(z_c = 1|y)$.

Theorem 4.3 (Target Function Approximation Error Bound). *Given a dataset $\mathcal{D} \sim (\mathbb{P}_{y,z,q})^N$ (as defined in (4)), for the predictor $f^{*}_{y\mapsto z_c}$,*

- *the following holds with probability at least $1-\delta$ for any $\delta \in (0,1)$:*

$$\mathop{\mathbb{E}}_{y\sim\mathbb{P}_y} \left[|f^{*}_{y\mapsto z_c}(y) - \mathbb{P}_{z|y}(z_c = 1|y)|^2\right] \leq \frac{1}{N}\sum_{i=1}^{N} \left|(z^i)_c - f^{*}_{y\mapsto z_c}(y^i)\right|^2 + \frac{4}{\sqrt{N}} + \sqrt{\frac{\log(1/\delta)}{2N}}. \tag{51}$$

- *under Assumption 1 and Assumption 2, the following holds with probability at least* $1-\delta$ *for any* $\delta \in (0,1)$*:*

$$\mathop{\mathbb{E}}_{y\sim\mathbb{P}_y}\left[|f^*_{y\mapsto z_c}(y) - \mathbb{P}_{z|y}(z_c=1|y)|^2\right] \leq \frac{4}{\sqrt{N}} + \sqrt{\frac{\log(1/\delta)}{2N}}. \tag{52}$$

Proof. The proof is provided in Appendix G. □

Lemma 4.4 (Practical Significance of Theorem 4.3). *The practical significance of Theorem 4.3 is for approximating the target function. Theorem 4.3 allows us to make the following assumption:*

$$f^*_{y\mapsto z_c}(y) \approx \mathbb{P}_{z|y}(z_c=1|y). \tag{53}$$

Using (44) and (18) in (53), we get

$$\min_{q\in\{1,2,\cdots,Q\}} \Gamma_{\mathcal{A}_{\mathbf{Y}^{c,q}}}(y) \approx -p\log\left(\mathbb{P}_{z|y}(z_c=1|y)\right). \tag{54}$$

Theorem 4.3 further allows us to determine the sample complexity for $f^*_{y\mapsto z_c}$, as stated in the Lemma 4.5:

Lemma 4.5 (Sample Complexity for $f^*_{y\mapsto z_c}$). *The number of data points, needed to be sampled from a distribution* $\mathbb{P}_{y,z,q}$ *to guarantee*

$$\left|\mathop{\mathbb{E}}_{y\sim\mathbb{P}_y}\left[|f^*_{y\mapsto z_c}(y) - \mathbb{P}_{z|y}(z_c=1|y)|^2\right] - \frac{1}{N}\sum_{i=1}^{N}\left|f^*_{y\mapsto z_c}(y^i) - (z^i)_c\right|^2\right| \leq \epsilon \tag{55}$$

with probability at least $1-\delta$ *for any* $\delta\in(0,1)$ *and* $\epsilon>0$*, is given as*

$$N(\epsilon,\delta) = \Omega\left(\frac{1}{\epsilon^2}\left(4+\sqrt{\frac{\log(1/\delta)}{2}}\right)^2\right). \tag{56}$$

Figure 3 plots the lower bound on sample complexity against target function approximation risk bound for two different values of failure probabilities.

Remark 9 (A Discussion on Sample Complexity). *Lemma 4.5 provides a generalized expression for sample complexity that is independent of the data distributions, yielding a conservative estimate. In practice, however, for any given (albeit unknown) data distribution, considerably fewer samples may suffice to achieve a specific risk value. This is demonstrated through an experiment on a toy example, in which N number of data samples* $(y^i\in[-1,1], z^i_1\in\{0,1\})_{i=1}^N$ *are generated as follows:*

$$y^i = -1 + \left(\frac{2}{N-1}\right)(i-1) \tag{57}$$

$$z^i_1 = \begin{cases} 1, & \text{if } |y^i| < 0.5, \\ 0, & \text{otherwise.} \end{cases} \tag{58}$$

In this toy example, the target function is defined as follows:

$$\mathbb{P}_{z|y}(z_1=1|y) = \begin{cases} 1, & \text{if } |y| < 0.5, \\ 0, & \text{otherwise.} \end{cases} \tag{59}$$

The target function approximation risk (defined as the left-hand side of inequality (55)) is experimentally evaluated by approximating the expected value using 50,000 equidistant points between -1 and 1. Figure 4 presents a plot comparing both N (the number of samples) and the sample complexity lower bound against the experimentally evaluated risk in the toy example. The conservative nature of the generalized sample complexity lower bound is

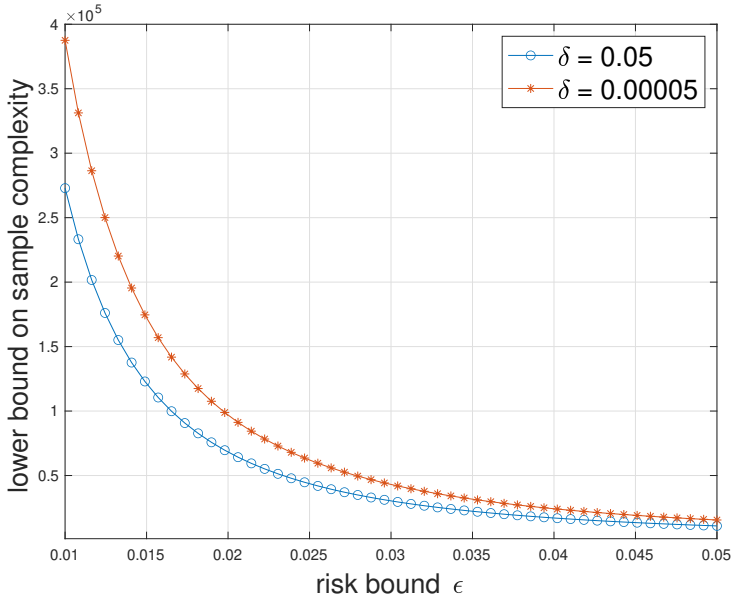


Fig. 3. A plot of the lower bound on sample complexity against target function approximation risk bound.

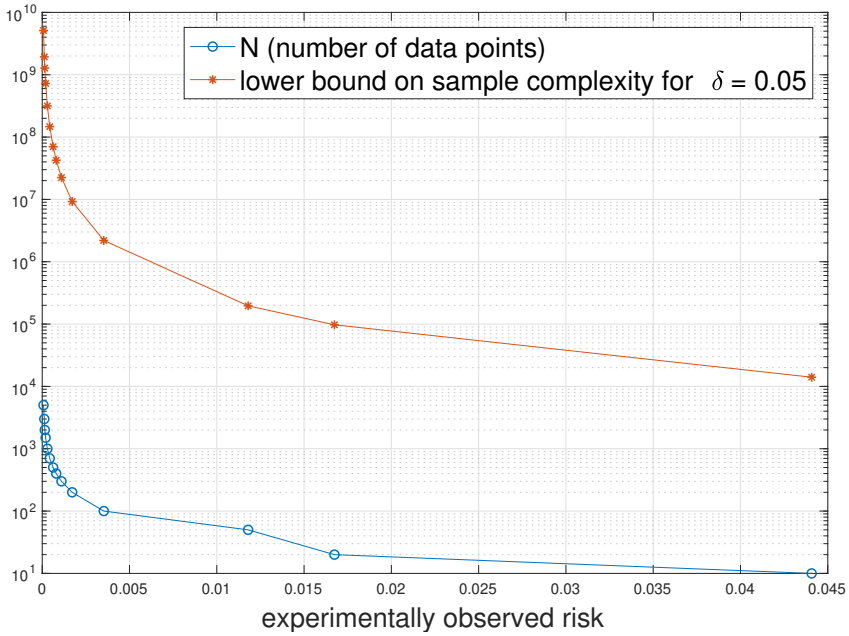


Fig. 4. The plot displays the number of samples (N) versus the observed risk in the toy example experiment, in comparison with the corresponding sample complexity lower bound.

clearly evident, as the experiments require far fewer samples to achieve a given risk value than what is estimated by Lemma 4.5.

Theorem 4.6 (Deterministic Analysis of Predictor $f^*_{y\mapsto z_c}$). *The degree of membership of a point $y \in \mathbb{R}^p$ to the c-th class, assigned by the predictor $f^*_{y\mapsto z_c}$ under Assumption 1, is related to the distance of that point from the c-th class labelled training samples as in the following:*

$$f^*_{y\mapsto z_c}(y) > \exp\left(-\left(\frac{1}{p}+\frac{\left|\mathcal{I}^{c,q^*(y)}\right|^2}{2\left\|\Upsilon^{c,q^*(y)}\right\|_F^2}\right)\left\|\left[\begin{array}{ccc} y-y^{\mathrm{I}_1^{c,q^*(y)}} & \cdots & y-y^{\mathrm{I}_{|\mathcal{I}^{c,q^*(y)}|}^{c,q^*(y)}}\end{array}\right]\right\|_2\right), \tag{60}$$

where

$$q^*(y) = \underset{q\in\{1,2,\cdots,Q\}}{\operatorname{argmin}} \Gamma_{\mathcal{A}_{\Upsilon^{c,q}}}(y). \tag{61}$$

Proof. The proof is provided in Appendix H. □

Remark 10 (Significance of Theorem 4.6). *Theorem 4.6 implies that if a point y is close to the c-th class labelled and $q^*(y)$-th client owned training samples $\left\{y^{\mathrm{I}_1^{c,q^*(y)}},\cdots,y^{\mathrm{I}_{|\mathcal{I}^{c,q^*(y)}|}^{c,q^*(y)}}\right\}$ (i.e. $\left\|\left[\begin{array}{ccc} y-y^{\mathrm{I}_1^{c,q^*(y)}} & \cdots & y-y^{\mathrm{I}_{|\mathcal{I}^{c,q^*(y)}|}^{c,q^*(y)}}\end{array}\right]\right\|_2$ is small), then the value $f^*_{y\mapsto z_c}(y)$ cannot be small. Thus, a small value of $f^*_{y\mapsto z_c}(y)$ indicates that y is at a far distance from the c-th class labelled training samples of all clients, and thus an interpretation of the predictor can be given in terms of the distance from training samples.*

4.4 Classification Applications in Federated Setting

Definition 4.7 (A Global Classifier). Given a distributed dataset $\mathcal{D} \sim (\mathbb{P}_{y,z,q})^N$ (as defined in (4)), a global classifier, $C : \mathbb{R}^p \to \{1, 2, \cdots, C\}$, is defined as

$$C(y) = \underset{c\in\{1,2,\cdots,C\}}{\operatorname{argmax}} \mathbb{P}_{z|y}(z_c = 1|y). \tag{62}$$

Using (54), $C(y)$ can be approximated as

$$\widehat{C}(y) = \underset{c\in\{1,2,\cdots,C\}}{\operatorname{argmin}} \left(\min_{q\in\{1,2,\cdots,Q\}} \Gamma_{\mathcal{A}_{\Upsilon^{c,q}}}(y)\right). \tag{63}$$

A local classifier is derived from the global classifier by staying confined to the local data as in Definition 4.8:

Definition 4.8 (A Local Classifier). For the q-th client with data $\{y^i \mid i \in \cup_{c=1}^{C}\mathcal{I}^{c,Q}\}$, the local classifier, $\widehat{C}_q : \mathbb{R}^p \to \{1, 2, \cdots, C\}$, is defined as

$$\widehat{C}_q(y) = \underset{c\in\{1,2,\cdots,C\}}{\operatorname{argmin}} \Gamma_{\mathcal{A}_{\Upsilon^{c,q}}}(y). \tag{64}$$

Remark 11 (Local Data with Missing Classes). *If the q-th client has zero c-th class labelled samples, then (63) is evaluated taking $\Gamma_{\mathcal{A}_{\Upsilon^{c,q}}}(y) = \infty$.*

Remark 12 (Addressing Computational Challenge of Big Datasets). *The computational challenge of KAHM modelling of big datasets can be addressed, as suggested by [26], by partitioning the total dataset into subsets*

and modelling each subset through a separate KAHM. Specifically, if $|\mathcal{I}^{c,q}|$ (i.e. the number of c-th class labelled samples that are owned by client q, that equals the number of rows in matrix $\mathbf{Y}^{c,q}$) is more than 1000, then we define

$$S = \lceil |\mathcal{I}^{c,q}|/1000 \rceil \tag{65}$$

$$\{\mathbf{Y}_1^{c,q}, \cdots, \mathbf{Y}_S^{c,q}\} = \text{clustering}\left(\{(\mathbf{Y}^{c,q})_{1,:}, \cdots, (\mathbf{Y}^{c,q})_{|\mathcal{I}^{c,q}|,:}\}, S\right) \tag{66}$$

$$\Gamma_{\mathcal{A}_{\mathbf{Y}^{c,q}}}(y) := \min_{s\in\{1,2,\cdots,S\}} \Gamma_{\mathcal{A}_{\mathbf{Y}_s^{c,q}}}(y). \tag{67}$$

That is, the total data samples (stored in the rows of matrix $\mathbf{Y}^{c,q}$) are partitioned into a number of subsets S (where S equals the rounding of $|\mathcal{I}^{c,q}|/1000$ towards the nearest integer) through clustering, and each subset is modelled through a separate KAHM resulting in a set of KAHMs $\{\mathcal{A}_{\mathbf{Y}_1^{c,q}}, \cdots, \mathcal{A}_{\mathbf{Y}_S^{c,q}}\}$, which are finally aggregated through the distance measure given in (67).

It is important to note that the proposed learning solution and its theoretical performance guarantees are derived without imposing any statistical assumptions on clients' data distributions. Consequently, large datasets can be partitioned into subsets and processed in parallel in a computationally efficient manner. This independence from data distribution assumptions not only broadens the applicability of the solution across diverse scenarios but also facilitates efficient handling of big data by leveraging parallel computing architectures. This design choice ultimately enables faster processing and improved scalability in real-world applications.

Remark 13 (Practical Significance of Global Classifier (63)). *The significance of the global classifier (63) is that its evaluation does not require individual KAHMs (that are owned by different clients), but only the distance measures, giving rise to a collaborative learning scheme as shown in Figure 5. Concretely, the collaborative learning scheme sketched in Figure 5 requires a processing of user inputs by client's local models. The passing of user inputs to an arbitrary client for local inference can be avoided by transferring all of the local KAHMs $\{\{\mathcal{A}_{\mathbf{Y}^{c,q}}\}_{c=1}^{C}\}_{q=1}^{Q}$ to the server.*

Remark 14 (Integration of Privacy-Enhancing Methodologies). *The proposed collaborative learning solution allows for a seamless integration of privacy-enhancing methodologies such as differential privacy and fully homomorphic encryption. For instance, differentially private collaborative learning is achieved via KAHM-based fabrication of privacy-preserving training data for each client [26]. Secure collaborative learning is enabled through fully homomorphic encryption of local model inferences followed by homomorphic evaluation of the global model [23, 29].*

5 Experiments

Existing research [26] has shown not only the privacy-preserving potential of KAHMs (by fabricating privacy-preserving data) but also that they remain computationally practical, given that they are capable of addressing computational challenges of big data (as stated in Remark 12). Thus, experiments to establish the privacy-preserving property and computational efficiency are not repeated here. However, Assumption 1 made here to derive a learning solution (cf. Theorem 4.1) still needs to be validated through experiments. Consequently, we have conducted experiments to this effect (see Section 5.1). Further, the competitive advantage in terms of performance of a KAHM-based approach to collaborative learning in a federated setting still needs to be established by comparing it to the state of the art methods. Federated learning experiments are provided in Section 5.2, followed by experiments of knowledge transfer across clients in Section 5.3. Finally, in Section 5.4, we show the effectiveness of our method in the single-class data scenario.

Implementation. The method was implemented using MATLAB (R2024a) and the source code was made publicly available on https://github.com/software-competence-center-hagenberg/GIKM. The experiments were performed on an iMac (M1, 2021) machine with 8 GB RAM. It is worth mentioning that the proposed method does not

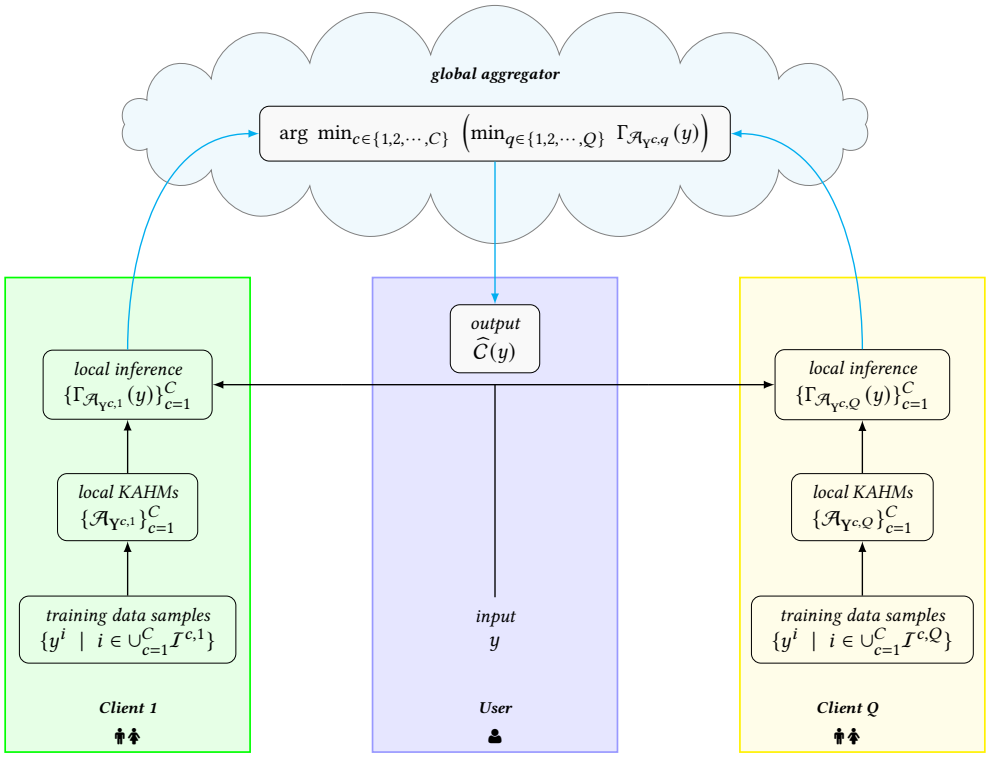


Fig. 5. A representation of the collaborative learning solution. For practical implementation, the local KAHMs $\{\{\mathcal{A}_{\mathbf{Y}^{c,q}}\}_{c=1}^{C}\}_{q=1}^{Q}$ can be transferred to the server to avoid the passing of user's input to any client for local inference. *The solution of global model learning problem has been derived without making any statistical assumptions regarding clients' data distributions, thereby providing by design a robustness towards statistical heterogeneity.*

involve any free parameters to be tuned, since building a KAHM requires only selecting the subspace dimension $n \leq p$, which is determined as stated in Appendix B. The computational challenge of big datasets is addressed as stated in Remark 12. Having defined client and class specific KAHMs (in accordance to Definition 2.1) from the available training data samples, no additional computational algorithm is required for the inference of both global (63) and local classifiers (64).

Datasets. We study the multi-class classification problem on the benchmark datasets including MNIST, Freiburg Groceries, Fashion MNIST, CIFAR-10, CIFAR-100, and Office-Caltech-10 datasets. The MNIST dataset contains 28×28 sized images (of digits) divided into 60000 training images and 10000 test images. The Freiburg Groceries dataset [14] has 4947 images of grocery products (commonly sold in Germany) labelled into 25 different categories and divided into 3929 training images and 1018 test images. The Fashion MNIST dataset contains 60000 training and 10000 test 28×28 grayscale images of fashion products from 10 categories. The CIFAR-10 dataset contains 50000 training and 10000 test 32×32 color images from 10 different classes. The CIFAR-100 dataset consists of 60000 32×32 color images from 100 classes with 500 training images and 100 test images in each class. The Office-Caltech-10 dataset, containing the 10 overlapping categories between the Office dataset and Caltech256

Table 1. Assessment of Assumption 1 and Assumption 2 on MNIST, Freiburg Groceries, CIFAR-10, and CIFAR-100 datasets.

dataset	MNIST	Freiburg Groceries	CIFAR-10	CIFAR-100
E1	0.0045	0.0039	0.0055	0.0049
E2	0.0996	0.0392	0.0995	0.0099
Accuracy	0.9870	0.8969	0.9122	0.7471

dataset, consists of images coming from four data sources: Amazon (958 images), Caltech (1123 images), DSLR (157 images), and Webcam (295 images).

Data Processing. KAHMs are built from training data samples. Since our experiments are on the images, a feature vector needs to be extracted from each image so that the client and class specific KAHMs could be built from the extracted feature vectors. For MNIST and Fashion MNIST datasets, 28×28 pixel values of each image are divided by 255 (to scale the values in the range from 0 to 1) and flattened to an equivalent 784−dimensional data point. For Freiburg Groceries, CIFAR-10, CIFAR-100, and Office-Caltech-10 datasets, the existing ResNet-50 neural network is employed as feature extractor by using the activations of "avg_pool" layer (i.e. the last average pooling layer just before the fully connected layer) as features, resulting into a 2048−dimensional data point for each image. For Office-Caltech10 images in transfer learning experiments, additionally a 4096-dimensional data point is computed from the activations of "fc6" layer of the existing VGG-16 neural network to compare the results with previous studies using same features. Finally, for all the datasets, the hyperbolic tangent function operates along each dimension of a data point to constrain the values between -1 and +1, resulting in the feature vectors to be considered for classification.

5.1 Validation of the Underlying Assumptions

Since the proposed solution has been derived (in Theorem 4.1) under Assumption 1, it is important to validate it experimentally. To check the validity of Assumption 1 in the experiments, the following score is defined:

$$E1 = \max_{c \in \{1,\cdots,C\}} \max_{i \in \bigcup_{q=1}^{Q} \mathcal{I}^{c,q}} \left| 1 - \exp\left(-\frac{\Gamma_{\mathcal{G}_c}(y^i)}{p} \right) \right|.$$

A low value of $E1$ (close to zero) will indicate the validity of Assumption 1, whereas if the converse is true our assumption does not hold. Assumption 2 was introduced to simplify our theoretical analysis. To check its validity experimentally, we define the following score:

$$E2 = \operatorname*{mean}_{c \in \{1,\cdots,C\},\ i \notin \bigcup_{q=1}^{Q} \mathcal{I}^{c,q}} \exp\left(-\frac{\Gamma_{\mathcal{G}_c}(y^i)}{p} \right). \tag{68}$$

An $E2$ value approximating zero confirms the validity of Assumption 2. One of the key advantages of our approach is its ability to accommodate the heterogeneity in clients' data distributions, even in the extreme case where no two clients share a class. To underscore this advantage, we examine this extreme case in experiments on benchmark datasets, where Assumption 1 and Assumption 2 are assessed for their validity. A single-class data scenario has been created assuming that training data samples of a class are completely owned by a single client. Thus, the number of clients is equal to the number of classes. Since Assumptions 1 and 2 concern global KAHMs

Table 2. Comparison of the averaged (over clients) test data accuracies obtained by our proposed method with previously available results [47] for different datasets in the non-iid label skew 20% federated learning experiment. The mean value of averaged accuracy over 3 independent runs of the experiment is reported.

Method	Fashion MNIST	CIFAR-10	CIFAR-100
KAHM Global Classifier (63)	**95.24**	**96.32**	**82.4**
FedAvg	77.3	49.8	53.73
FedProx	74.9	50.7	54.35
FedNova	70.4	46.5	53.61
SCAFFOLD	42.8	49.1	54.15
KAHM Local Classifier (64)	**97.95**	**97.82**	**79.88**
LG-FedAvg	96.8	86.31	45.98
Per-FedAvg	95.95	85.46	60.19
IFCA	97.15	87.99	71.84
CFL	77.93	51.11	40.29
PACFL	97.54	89.3	73.10

that mitigate the effects of sample distribution across clients, any setting of sample distribution, including the current one, is sufficient to assess these assumptions.

The observed $E1$ value and accuracy obtained by the global classifier (63) for the different datasets are provided in Table 1. The low values of $E1$, with $E1 < 0.01$, across all of the considered datasets validate Assumption 1 and thus the proposed solution is justified.

Table 1 reports the $E2$ values obtained from experiments on various datasets. The fact that $E2$ does not exceed 0.0996 in any of the cases supports the use of Assumption 2 for simplifying our theoretical analysis. It is important to note that validating Assumption 2 is required only for Theorem 4.2 and inequality (52). The primary contribution of this paper is the development of a learning solution that operates independently of Assumption 2.

5.2 Federated Learning

Following [47], we consider a non-iid label skew 20% (or 30%) federated learning setting, in which

- the number of clients is equal to 100;
- each client is first randomly assigned 20% (or 30%) of the total available class-labels in a dataset, and then the training samples of each class are randomly distributed equally among clients who have been assigned that class;
- all of the test samples of a class are assigned to every client who has been assigned that class;
- the accuracy over the client's test data, averaged across clients, is calculated to evaluate the performance.

The proposed KAHM-based federated learning approach, unlike the state of the art federated learning methods, is outside the realm of gradient descent-based learning of parametric neural networks. Therefore, the performance of the proposed method is evaluated and compared with previously available results [47] of existing methods. Specifically, the proposed global classifier (63) is compared with the methods (that train a single global model across all clients): FedAvg [38], FedProx [33], FedNova [48], and SCAFFOLD [18]. The proposed local classifier (64) is compared with personalised federated learning methods: LG-FedAvg [37], Per-FedAvg [3], CFL [43], IFCA [6], and PACFL [47]. Table 2 reports the results for non-iid label skew 20% and results for non-iid label skew 30% are reported in Table 3.

Table 3. Comparison of the averaged (over clients) test data accuracies obtained by our proposed method with the previously available results [47] for different datasets in the non-iid label skew 30% federated learning experiment. The mean value of averaged accuracy over 3 independent runs of the experiment is reported.

Method	Fashion MNIST	CIFAR-10	CIFAR-100
KAHM Global Classifier (63)	**92.75**	**95.06**	**80.12**
FedAvg	80.7	58.3	54.73
FedProx	82.5	57.1	53.31
FedNova	78.9	54.4	54.62
SCAFFOLD	77.7	57.8	54.9
KAHM Local Classifier (64)	**95.51**	**95.90**	**73.44**
LG-FedAvg	94.21	76.58	35.91
Per-FedAvg	92.87	77.67	56.42
IFCA	95.22	80.95	67.39
CFL	78.44	52.57	35.23
PACFL	95.46	82.77	67.71

Table 4. A summary of the results obtained by different methods in semi-supervised transfer learning experiments on the Office-Caltech-10 dataset. The mean and standard deviation of averaged accuracy over 12 different transfer learning experiments are reported.

Method	**Feature Type**	**Accuracy (%)**	**Rank**
KAHM Global Classifier (63)	RESNET50	**94.3 ± 3.9**	**1**
KAHM Global Classifier (63)	VGG-FC6	93.8 ± 3.9	2
CDMMA	VGG-FC6	88.4 ± 4.3	3
ILS (1-NN)	VGG-FC6	88.4 ± 4.4	4
CDLS	VGG-FC6	85.9 ± 4.9	5
MMDT	VGG-FC6	80.8 ± 4.9	7
HFA	VGG-FC6	83.7 ± 5.3	6
OBTL	SURF	58.9 ± 14.6	8
ILS (1-NN)	SURF	55.6 ± 12.0	9
CDLS	SURF	53.5 ± 13.0	10
MMDT	SURF	52.5 ± 13.7	11
HFA	SURF	48.1 ± 12.0	12

It can be seen from Table 2 and Table 3 that our KAHM-based approach consistently achieves considerably better performance, in other words, it visibly outperforms the state of the art federated learning methods. In particular, the KAHM global classifier improved the best existing performance on CIFAR-100 by +9.3% (in the non-iid label skew 20% scenario) and by +12.41% (in the non-iid label skew 30% scenario). Similarly, the KAHM global classifier improved the best existing performance on CIFAR-10 by +7.02% (in the non-iid label skew 20% scenario) and by +12.29% (in the non-iid label skew 30% scenario).

Table 5. Results of single-class data federated learning experiments on 5 different train-test splits of Freiburg groceries data

methods	accuracy (in %) on test images					
	1	2	3	4	5	mean
KAHM Global Classifier (63)	**89.69**	**87.79**	**88.02**	**88.44**	**87.99**	**88.39**
membership-mappings [24]	87.82	87.06	85.88	85.63	86.19	86.52
nonparametric fuzzy image mapping [22]	88.21	86.64	85.36	85.13	85.79	86.23
Gaussian fuzzy-mapping [28]	83.50	81.52	79.73	79.60	80.48	80.97
SVM [24]	77.90	79.54	77.17	76.98	76.98	77.71
1-NN [24]	78.00	77.97	77.38	76.58	76.28	77.24
back-propagation training of a deep network [24]	75.25	77.24	72.67	73.37	71.57	74.02
2-NN [24]	73.48	73.38	70.11	70.05	70.57	71.52
4-NN [24]	72.50	73.39	68.89	71.16	70.87	71.36
Random Forest [24]	63.17	62.63	59.47	59.50	59.76	60.90
Naive Bayes [24]	56.78	56.78	53.74	55.08	56.26	55.73
Ensemble Learning [24]	38.31	39.35	38.89	37.69	38.34	38.51
Decision Tree [24]	31.34	30.59	32.14	31.06	30.73	31.17

5.3 Transfer Learning in a Federated Setting

To evaluate the performance of the proposed KAHM-based approach to collaborative learning in a federated setting, we study the performance of global classifier (63) in transferring knowledge from a client (corresponding to the source domain) to another client (corresponding to the target domain). For this, we consider the Office-Caltech-10 dataset consisting of four domains: Amazon, Caltech, DSLR, and Webcam. This dataset has been widely used in the literature, e.g., [9, 7, 17, 8, 21], for evaluating multi-class accuracy performance in a standard domain adaptation setting with a small number of labelled target samples. We follow the experimental setup of previous studies such as [9, 7, 17, 8, 21] on semi-supervised transfer learning using the Office-Caltech-10 dataset:

- the number of training samples per class in the source domain is 20 for Amazon and is 8 for Caltech, DSLR, and Webcam;
- the number of labelled samples per class in the target domain is 3 for all the four domains;
- 20 random train/test splits are created and the performance on target domain test samples is averaged over 20 experiments.

We consider the following existing method for a comparison: Invariant Latent Space ILS (1-NN) [7], Cross-Domain Landmark Selection CDLS [45], Maximum Margin Domain Transform MMDT [8], Heterogeneous Feature Augmentation HFA [34], Optimal Bayesian Transfer Learning OBTL [17], and Conditionally Deep Membership-Mapping Autoencoder CDMMA [21]. For a fair comparison, the proposed method is also studied with the deep-net VGG-FC6 features extracted from the images, as in the previous studies [7, 21]. The Office-Caltech-10 dataset has also been previously studied using SURF features [9, 7, 17, 8], and thus existing results using SURF features are additionally considered for a comparison. Taking a domain as source and other domain as target, 12 different transfer learning experiments can be performed on the 4 domains of Office-Caltech-10 dataset. The results of all 12 experiments have been summarised in Table 4.

The best performance of our proposed method is observed in Table 4. Overall, as observed from Table 4, the KAHM global classifier improved the best existing performance on Office-Caltech-10 dataset by +5.9% using ResNet-50 features and by +5.4% using VGG16 features.

5.4 Single-Class Data

The Freiburg Groceries dataset is again considered under the scenario that a client owns training samples of only one class and all of the samples of that class, resulting in $Q = C$. As in the previous studies on this dataset [24, 22, 28], the experiments are performed for 5 different train-test splits of data. The obtained results are reported in Table 5 and compared with the previous results on this dataset. The improved performance shown in Table 5 proves that the proposed method remains competitive under the considered single-class data scenario.

6 Conclusion

Collaborative learning in federated setting is becoming popular due to the increasingly distributed nature of data across a variety of domains. Overall, we provide a KAHM-based comprehensive solution. Unlike most of the existing methods, our approach does not require the multiple rounds of communication between clients and server for the learning of the global model, does not require clients to perform multiple epochs of local optimisation using stochastic gradient descent, and does not require any tuning of the free parameters. This paper provides a new theory for KAHM-based collaborative learning in a federated setting. Our work sheds light on the theoretical understanding of KAHM-based collaborative learning in federated settings and provides insights for designing a suitable learning solution. We have provided theoretical guarantees for the proposed solution and empirically demonstrated its effectiveness by showing a considerable improvement from the existing results on benchmark datasets in federated learning and transfer learning experiments. An interpretation and justification for the proposed solution is also provided in terms of a distance measure from training data points.

Acknowledgments

The research reported in this paper has been supported by the state of Upper Austria as part of #upperVISION2030 under the Secure Prescriptive Analytics (SPA) project; by Austrian Research Promotion Agency FFG under ARIKI project; by the Federal Ministry for Innovation, Mobility and Infrastructure (BMIMI), the Federal Ministry for Economy, Energy and Tourism (BMWET), and the State of Upper Austria in the frame of the SCCH competence center INTEGRATE [(FFG grant no. 892418)] in the COMET - Competence Centers for Excellent Technologies Programme managed by Austrian Research Promotion Agency FFG. Bowles is partially supported by the Austrian Funding Council (FWF) under Meitner M 3338-N.

References

[1] M. Belkin, S. Ma, and S. Mandal. 2018. To understand deep learning we need to understand kernel learning. In *Proceedings of the 35th International Conference on Machine Learning* (Proceedings of Machine Learning Research). J. Dy and A. Krause, (Eds.) Vol. 80. PMLR, (Oct. 2018), 541–549.

[2] L. Collins, H. Hassani, A. Mokhtari, and S. Shakkottai. 2021. Exploiting shared representations for personalized federated learning. In *Proceedings of the 38th International Conference on Machine Learning* (Proceedings of Machine Learning Research). M. Meila and T. Zhang, (Eds.) Vol. 139. PMLR, (18–24 Jul 2021), 2089–2099.

[3] A. Fallah, A. Mokhtari, and A. Ozdaglar. 2020. Personalized federated learning with theoretical guarantees: a model-agnostic meta-learning approach. In *Advances in Neural Information Processing Systems*. H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, (Eds.) Vol. 33. Curran Associates, Inc., 3557–3568.

[4] P. M. Ghari and Y. Shen. 2022. Personalized online federated learning with multiple kernels. In *Advances in Neural Information Processing Systems*. A. H. Oh, A. Agarwal, D. Belgrave, and K. Cho, (Eds.)

[5] B. Gholami and A. Hajisami. 2016. Kernel auto-encoder for semi-supervised hashing. In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 1–8.

[6] A. Ghosh, J. Chung, D. Yin, and K. Ramchandran. 2020. An efficient framework for clustered federated learning. In *Proceedings of the 34th International Conference on Neural Information Processing Systems* (NIPS '20) Article 1643. Curran Associates Inc., Red Hook, NY, USA, 12 pages. ISBN: 9781713829546.

[7] S. Herath, M. Harandi, and F. Porikli. 2017. Learning an invariant hilbert space for domain adaptation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (July 2017).

[8] J. Hoffman, E. Rodner, J. Donahue, B. Kulis, and K. Saenko. 2014. Asymmetric and category invariant feature transformations for domain adaptation. *International Journal of Computer Vision*, 109, 1, 28–41. DOI: 10.1007/s11263-014-0719-3.

[9] J. Hoffman, E. Rodner, J. Donahue, K. Saenko, and T. Darrell. 2013. Efficient learning of domain-invariant image representations. *CoRR*, abs/1301.3224.

[10] T. Hofmann, B. Schölkopf, and A. J. Smola. 2008. Kernel methods in machine learning. *The Annals of Statistics*, 36, 3, 1171–1220.

[11] S. Hong and J. Chae. 2022. Communication-efficient randomized algorithm for multi-kernel online federated learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44, 12, 9872–9886.

[12] B. Huang, X. Li, Z. Song, and X. Yang. 2021. Fl-ntk: a neural tangent kernel-based framework for federated learning analysis. In *Proceedings of the 38th International Conference on Machine Learning* (Proceedings of Machine Learning Research). M. Meila and T. Zhang, (Eds.) Vol. 139. PMLR, (18–24 Jul 2021), 4423–4434.

[13] A. Jacot, F. Gabriel, and C. Hongler. 2018. Neural tangent kernel: convergence and generalization in neural networks. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems* (NIPS'18). Curran Associates Inc., Montréal, Canada, 8580–8589.

[14] P. Jund, N. Abdo, A. Eitel, and W. Burgard. 2016. The freiburg groceries dataset. *CoRR*, abs/1611.05799.

[15] P. Kairouz et al. 2021. Advances and open problems in federated learning. *Foundations and trends® in machine learning*, 14, 1–2, 1–210.

[16] M. Kampffmeyer, S. Løkse, F. M. Bianchi, R. Jenssen, and L. Livi. 2018. The deep kernelized autoencoder. *Applied Soft Computing*, 71, 816–825.

[17] A. Karbalayghareh, X. Qian, and E. R. Dougherty. 2018. Optimal bayesian transfer learning. *IEEE Transactions on Signal Processing*, 66, 14, 3724–3739.

[18] S. P. Karimireddy, S. Kale, M. Mohri, S. Reddi, S. Stich, and A. T. Suresh. 2020. SCAFFOLD: stochastic controlled averaging for federated learning. In *Proceedings of the 37th International Conference on Machine Learning* (Proceedings of Machine Learning Research). H. D. III and A. Singh, (Eds.) Vol. 119. PMLR, (13–18 Jul 2020), 5132–5143.

[19] A. Khaled, K. Mishchenko, and P. Richtarik. 2020. Tighter theory for local sgd on identical and heterogeneous data. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics* (Proceedings of Machine Learning Research). S. Chiappa and R. Calandra, (Eds.) Vol. 108. PMLR, (26–28 Aug 2020), 4519–4529.

[20] M. Kumar, M. Rossbory, B. A. Moser, and B. Freudenthaler. 2021. An optimal $(\epsilon, \delta)-$differentially private learning of distributed deep fuzzy models. *Information Sciences*, 546, 87–120.

[21] M. Kumar. 2023. Differentially private transferrable deep learning with membership-mappings. *Advances in Computational Intelligence*, 3, 1, 1–27.

[22] M. Kumar and B. Freudenthaler. 2020. Fuzzy membership functional analysis for nonparametric deep models of image features. *IEEE Transactions on Fuzzy Systems*, 28, 12, 3345–3359.

[23] M. Kumar, B. Moser, and L. Fischer. 2023. Secure federated learning with kernel affine hull machines. In *31st European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, ESANN 2023, Bruges, Belgium, October 4-6, 2023*. https://doi.org/10.14428/esann/2023.ES2023-56.

[24] M. Kumar, B. Moser, L. Fischer, and B. Freudenthaler. 2021. Membership-mappings for data representation learning: a bregman divergence based conditionally deep autoencoder. In *Database and Expert Systems Applications - DEXA 2021 Workshops*. G. Kotsis et al., (Eds.) Springer International Publishing, Cham, 138–147.

[25] M. Kumar, B. Moser, L. Fischer, and B. Freudenthaler. 2021. Membership-mappings for data representation learning: measure theoretic conceptualization. In *Database and Expert Systems Applications - DEXA 2021 Workshops*. G. Kotsis et al., (Eds.) Springer International Publishing, Cham, 127–137.

[26] M. Kumar, B. A. Moser, and L. Fischer. 2024. On mitigating the utility-loss in differentially private learning: a new perspective by a geometrically inspired kernel approach. *Journal of Artificial Intelligence Research*, 79, 515–567.

[27] M. Kumar, M. Rossbory, B. A. Moser, and B. Freudenthaler. 2020. Differentially private learning of distributed deep models. In *Adjunct Publication of the 28th ACM Conference on User Modeling, Adaptation and Personalization* (UMAP '20 Adjunct). Association for Computing Machinery, Genoa, Italy, 193–200.

[28] M. Kumar, S. Singh, and B. Freudenthaler. 2021. Gaussian fuzzy theoretic analysis for variational learning of nested compositions. *International Journal of Approximate Reasoning*, 131, 1–29.

[29] M. Kumar, W. Zhang, L. Fischer, and B. Freudenthaler. 2023. Membership mappings for practical secure distributed deep learning. *IEEE Transactions on Fuzzy Systems*, 31, 8, 2617–2631.

[30] M. Kumar, W. Zhang, M. Weippert, and B. Freudenthaler. 2021. An explainable fuzzy theoretic nonparametric deep model for stress assessment using heartbeat intervals analysis. *IEEE Transactions on Fuzzy Systems*, 29, 12, 3873–3886.

[31] P. Laforgue, S. Clémençon, and F. d'Alché-Buc. 2019. Autoencoding any data through kernel autoencoders. In *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics* (Proceedings of Machine Learning Research). K. Chaudhuri and M. Sugiyama, (Eds.) Vol. 89. PMLR, (16–18 Apr 2019), 1061–1069.

[32] Q. Li, B. He, and D. Song. 2023. Adversarial collaborative learning on non-IID features. In *Proceedings of the 40th International Conference on Machine Learning* (Proceedings of Machine Learning Research). A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, and J. Scarlett, (Eds.) Vol. 202. PMLR, (23–29 Jul 2023), 19504–19526.

[33] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith. 2020. Federated optimization in heterogeneous networks. In *Proceedings of Machine Learning and Systems 2020, MLSys 2020, Austin, TX, USA, March 2-4, 2020*. I. S. Dhillon, D. S. Papailiopoulos, and V. Sze, (Eds.) mlsys.org.

[34] W. Li, L. Duan, D. Xu, and I. W. Tsang. 2014. Learning with augmented features for supervised and semi-supervised heterogeneous domain adaptation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36, 6, 1134–1148.

[35] X. Li, K. Huang, W. Yang, S. Wang, and Z. Zhang. 2020. On the convergence of fedavg on non-iid data. In *International Conference on Learning Representations*. https://openreview.net/forum?id=HJxNAnVtDS.

[36] X. Li, M. Jiang, X. Zhang, M. Kamp, and Q. Dou. 2021. FedBN: federated learning on non-IID features via local batch normalization. In *International Conference on Learning Representations*. https://openreview.net/forum?id=6YEQUn0QICG.

[37] P. P. Liang, T. Liu, Z. Liu, R. Salakhutdinov, and L. Morency. 2020. Think locally, act globally: federated learning with local and global representations. *CoRR*, abs/2001.01523. http://arxiv.org/abs/2001.01523.

[38] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y. Arcas. 2017. Communication-Efficient Learning of Deep Networks from Decentralized Data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics* (Proceedings of Machine Learning Research). A. Singh and J. Zhu, (Eds.) Vol. 54. PMLR, (20–22 Apr 2017), 1273–1282.

[39] M. Mohri, A. Rostamizadeh, and A. Talwalkar. 2018. *Foundations of Machine Learning*. (2nd ed.). *Adaptive Computation and Machine Learning*. MIT Press, Cambridge, MA. ISBN: 978-0-262-03940-6.

[40] N. K. Nikhitha, A. L. Afzal, and S. Asharaf. 2021. Deep kernel machines: a survey. *Pattern Anal. Appl.*, 24, 2, (May 2021), 537–556.

[41] Z. Qu, K. Lin, Z. Li, J. Zhou, and Z. Zhou. 2023. A unified linear speedup analysis of federated averaging and nesterov fedavg. *Journal of Artificial Intelligence Research*, 78, 1143–1200.

[42] A. Rudi, L. Carratino, and L. Rosasco. 2017. Falkon: an optimal large scale kernel method. In *Advances in Neural Information Processing Systems*. I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, (Eds.) Vol. 30. Curran Associates, Inc.

[43] F. Sattler, K.-R. Müller, and W. Samek. 2021. Clustered federated learning: model-agnostic distributed multitask optimization under privacy constraints. *IEEE Transactions on Neural Networks and Learning Systems*, 32, 8, 3710–3722.

[44] B. Sun, H. Huo, Y. Yang, and B. Bai. 2021. Partialfed: cross-domain personalized federated learning via partial initialization. In *Advances in Neural Information Processing Systems*. M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, (Eds.) Vol. 34. Curran Associates, Inc., 23309–23320.

[45] Y. H. Tsai, Y. Yeh, and Y. F. Wang. 2016. Learning cross-domain landmarks for heterogeneous domain adaptation. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 5081–5090.

[46] S. Vahidian, M. Morafah, C. Chen, M. Shah, and B. Lin. 2024. Rethinking data heterogeneity in federated learning: introducing a new notion and standard benchmarks. *IEEE Transactions on Artificial Intelligence*, 5, 3, 1386–1397. DOI: 10.1109/TAI.2023.3293068.

[47] S. Vahidian, M. Morafah, W. Wang, V. Kungurtsev, C. Chen, M. Shah, and B. Lin. 2023. Efficient distribution similarity identification in clustered federated learning via principal angles between client data subspaces. In *Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence and Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence and Thirteenth Symposium on Educational Advances in Artificial Intelligence* (AAAI'23/IAAI'23/EAAI'23) Article 1128. AAAI Press, 10 pages. ISBN: 978-1-57735-880-0.

[48] J. Wang, Q. Liu, H. Liang, G. Joshi, and H. V. Poor. 2020. Tackling the objective inconsistency problem in heterogeneous federated optimization. In *Proceedings of the 34th International Conference on Neural Information Processing Systems* (NIPS '20) Article 638. Curran Associates Inc., Red Hook, NY, USA, 13 pages.

[49] A. G. Wilson, Z. Hu, R. Salakhutdinov, and E. P. Xing. 2016. Deep kernel learning. In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics* (Proceedings of Machine Learning Research). A. Gretton and C. C. Robert, (Eds.) Vol. 51. PMLR, Cadiz, Spain, (Sept. 2016), 370–378.

[50] M. Ye, X. Fang, B. Du, P. C. Yuen, and D. Tao. 2023. Heterogeneous federated learning: state-of-the-art and research challenges. *ACM Comput. Surv.*, 56, 3, Article 79, (Oct. 2023), 44 pages. DOI: 10.1145/3625558.

[51] Q. Zhang, J. Yang, W. Zhang, M. Kumar, J. Liu, J. Liu, and X. Li. 2023. Deep fuzzy mapping nonparametric model for real-time demand estimation in water distribution systems: a new perspective. *Water Research*, 241, 120145.

[52] W. Zhang, M. Kumar, W. Ding, X. Li, and J. Yu. 2022. Variational learning of deep fuzzy theoretic nonparametric model. *Neurocomputing*, 506, 128–145.

A Details of KAHM Definition 2.1

Given a finite number of samples: $\mathbf{Y} = \begin{bmatrix} y^1 & \cdots & y^N \end{bmatrix}^T$ with $y^1, \cdots, y^N \in \mathbb{R}^p$ and a subspace dimension $n \leq p$; a kernel affine hull machine $\mathcal{A}_{\mathbf{Y},n} : \mathbb{R}^p \to \text{aff}(\{y^1, \cdots, y^N\})$ is defined as

$$\mathcal{A}_{\mathbf{Y},n}(y) := \frac{h^1_{k_\theta,\mathbf{YP}^T,\lambda^*}(\mathbf{P}y)}{\sum_{i=1}^N h^i_{k_\theta,\mathbf{YP}^T,\lambda^*}(\mathbf{P}y)} y^1 + \cdots + \frac{h^N_{k_\theta,\mathbf{YP}^T,\lambda^*}(\mathbf{P}y)}{\sum_{i=1}^N h^i_{k_\theta,\mathbf{YP}^T,\lambda^*}(\mathbf{P}y)} y^N. \tag{69}$$

Here,

- $\mathbf{P} \in \mathbb{R}^{n\times p}$ $(n \leq p)$ is an encoding matrix such that product $\mathbf{P}y$ is a lower-dimensional encoding for y. For a given subspace dimension n, $\mathbf{P}$ is defined by setting the i-th row of $\mathbf{P}$ as equal to transpose of eigenvector corresponding to i-th largest eigenvalue of sample covariance matrix of dataset $\{y^1, \cdots, y^N\}$.
- $k_\theta : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a positive-definite real-valued kernel on $\mathcal{X}$ with a corresponding reproducing kernel Hilbert space $\mathcal{H}_{k_\theta}(\mathcal{X})$ where

  $$\mathcal{X} = \{\mathbf{P}y \mid y \in \mathbb{R}^p\}. \tag{70}$$

  The kernel function k_θ is chosen of Gaussian type:

  $$k_\theta(x^i, x^j) := \exp\left(-\frac{(x^i - x^j)^T\theta^{-1}(x^i - x^j)}{2n}\right), \tag{71}$$

  where $\theta \succ 0$ is sample covariance matrix of dataset $\{\mathbf{P}y^1, \cdots, \mathbf{P}y^N\}$.
- The function $h^i_{k_\theta,\mathbf{YP}^T,\lambda} : \mathcal{X} \to \mathbb{R}$, such that $h^i_{k_\theta,\mathbf{YP}^T,\lambda} \in \mathcal{H}_{k_\theta}(\mathcal{X})$, approximates the indicator function $\mathbb{1}_{\{\mathbf{P}y^i\}} : \mathcal{X} \to \{0, 1\}$ as the solution of following kernel regularized least squares problem:

  $$h^i_{k_\theta,\mathbf{YP}^T,\lambda} = \arg \min_{f\in\mathcal{H}_{k_\theta}(\mathcal{X})} \left(\sum_{j=1}^N \left|\mathbb{1}_{\{\mathbf{P}y^i\}}(\mathbf{P}y^j) - f(\mathbf{P}y^j)\right|^2 + \lambda \|f\|^2_{\mathcal{H}_k(\mathcal{X})}\right),\ \lambda \in \mathbb{R}_+. \tag{72}$$

  The solution follows as

  $$h^i_{k_\theta,\mathbf{YP}^T,\lambda}(\cdot) = (\mathbf{I}_N)_{i,:}\left(\mathbf{K}_{\mathbf{YP}^T} + \lambda\mathbf{I}_N\right)^{-1}\begin{bmatrix} k_\theta(\cdot, \mathbf{P}y^1) & \cdots & k_\theta(\cdot, \mathbf{P}y^N)\end{bmatrix}^T \tag{73}$$

  where $(\mathbf{I}_N)_{i,:}$ denotes the i-th row of identity matrix of size N and $\mathbf{K}_{\mathbf{YP}^T}$ is $N \times N$ kernel matrix with its (i, j)-th element defined as

  $$(\mathbf{K}_{\mathbf{YP}^T})_{ij} := k_\theta(\mathbf{P}y^i, \mathbf{P}y^j). \tag{74}$$

  The value $h^i_{k_\theta,\mathbf{YP}^T,\lambda}(\mathbf{P}y)$ represents the kernel-smoothed membership of point Py to the set $\{\mathbf{P}y^i\}$.
- The regularization parameter $\lambda^* \in \mathbb{R}_+$ is given as

  $$\lambda^* = \hat{e} + \frac{2}{pN}\|\mathbf{Y}\|^2_F, \tag{75}$$

  where $\hat{e}$ is the unique fixed point of $f_{k_\theta,\mathbf{YP}^T,\mathbf{Y}}$ such that

  $$\hat{e} = f_{k_\theta,\mathbf{YP}^T,\mathbf{Y}}(\hat{e}, \frac{2}{pN}\|\mathbf{Y}\|^2_F), \tag{76}$$

  with $f_{k_\theta,\mathbf{YP}^T,\mathbf{Y}} : \mathbb{R}_+ \times \mathbb{R}_+ \to \mathbb{R}_+$ defined as

  $$f_{k_\theta,\mathbf{YP}^T,\mathbf{Y}}(e, \tau) := \frac{1}{pN}\sum_{j=1}^p \left\|(\mathbf{Y})_{:,j} - \mathbf{K}_{\mathbf{YP}^T}\left(\mathbf{K}_{\mathbf{YP}^T} + (e + \tau)\mathbf{I}_N\right)^{-1}(\mathbf{Y})_{:,j}\right\|^2. \tag{77}$$

The following iterations

$$e|_{it+1} = f_{k_\theta, \mathbf{Y}\mathbf{P}^T, \mathbf{Y}}(e|_{it}, \frac{2}{pN}\|\mathbf{Y}\|_F^2),\ it \in \{0, 1, \cdots\} \tag{78}$$

$$e|_0 \in (0, \frac{1}{pN}\|\mathbf{Y}\|_F^2) \tag{79}$$

converge to $\hat{e}$.

- The image of $\mathcal{A}_{\mathbf{Y},n}$ defines a region in the affine hull of $\{y^1, \cdots, y^N\}$. That is,

$$\mathcal{A}_{\mathbf{Y},n}[\mathbb{R}^p] := \{\mathcal{A}_{\mathbf{Y},n}(y) \ \mid \ y \in \mathbb{R}^p\} \subset \text{aff}(\{y^1, \cdots, y^N\}). \tag{80}$$

B Practical Choice for KAHM Subspace Dimension

Given N number of samples, the subspace dimension n can not exceed data dimension p and $N-1$ (as the number of principal components with non-zero variance cannot exceed $N-1$). Further, n should not be too high to cause negligible variance along any of the principal components. This can be ensured by checking data variance along each principal component, and if needed decrementing n by 1 till required. Following algorithm is suggested to practically determine n:

Require: Dataset $\{y^i \in \mathbb{R}^p\}_{i=1}^N$.

1: $n \leftarrow \min(20, p, N-1)$.
2: Define $\mathbf{P} \in \mathbb{R}^{n \times p}$ such that the i-th row of $\mathbf{P}$ is equal to transpose of eigenvector corresponding to i-th largest eigenvalue of sample covariance matrix of samples $\{y^1, \cdots, y^N\}$.
3: Define $x^i = \mathbf{P}y^i,\ \forall i \in \{1, 2, \cdots, N\}$.
4: **while** $\min_{1 \le j \le n} \left(\max_{1 \le i \le N}(x^i)_j - \min_{1 \le i \le N}(x^i)_j\right) < 1\text{e}{-3}$ **do**
5: $\quad n \leftarrow n - 1$.
6: $\quad$ Define $\mathbf{P} \in \mathbb{R}^{n \times p}$ such that the i-th row of $\mathbf{P}$ is equal to transpose of eigenvector corresponding to i-th largest eigenvalue of sample covariance matrix of dataset $\{y^1, \cdots, y^N\}$.
7: $\quad$ Define $x^i = \mathbf{P}y^i,\ \forall i \in \{1, 2, \cdots, N\}$.
8: **end while**
9: **return** n

C Proof of Theorem 3.1

Consider

$$\widehat{\mathcal{R}}_{\mathcal{D}}(\mathcal{M}_{\mathcal{D},c}) = \frac{1}{N}\mathop{\mathbb{E}}_{\sigma}\left[\sup_{f_{y \mapsto z_c} \in \mathcal{M}_{\mathcal{D},c}} \sum_{i=1}^N \sigma_i\, f_{y \mapsto z_c}(y^i)\right] \tag{81}$$

$$= \frac{1}{N}\mathop{\mathbb{E}}_{\sigma}\left[\sup_{f_{y \mapsto z_c} \in \mathcal{M}_{\mathcal{D},c}} \sum_{i=1}^N \sigma_i\ \left\langle f_{y \mapsto z_c}, \mathcal{K}_c(\cdot, y^i)\right\rangle_{\mathcal{H}_{\mathcal{K}_c}(\mathbb{R}^p)}\right] \tag{82}$$

$$= \frac{1}{N}\mathop{\mathbb{E}}_{\sigma}\left[\sup_{f_{y \mapsto z_c} \in \mathcal{M}_{\mathcal{D},c}} \left\langle f_{y \mapsto z_c}, \sum_{i=1}^N \sigma_i \mathcal{K}_c(\cdot, y^i)\right\rangle_{\mathcal{H}_{\mathcal{K}_c}(\mathbb{R}^p)}\right], \tag{83}$$

where (82) follows from the reproducing property of the kernel $\mathcal{K}_c$. Using Cauchy–Schwarz inequality,

$$\widehat{\mathcal{R}}_{\mathcal{D}}(\mathcal{M}_{\mathcal{D},c}) \le \frac{1}{N}\mathop{\mathbb{E}}_{\sigma}\left[\sup_{f_{y \mapsto z_c} \in \mathcal{M}_{\mathcal{D},c}} \|f_{y \mapsto z_c}\|_{\mathcal{H}_{\mathcal{K}_c}(\mathbb{R}^p)} \left\|\sum_{i=1}^N \sigma_i \mathcal{K}_c(\cdot, y^i)\right\|_{\mathcal{H}_{\mathcal{K}_c}(\mathbb{R}^p)}\right], \tag{84}$$

and using (28), we have

$$\widehat{\mathcal{R}}_{\mathcal{D}}(\mathcal{M}_{\mathcal{D},c}) \leq \frac{1}{N}\mathop{\mathbb{E}}_{\sigma}\left[\left\|\sum_{i=1}^{N}\sigma_i\mathcal{K}_c(\cdot,y^i)\right\|_{\mathcal{H}_{\mathcal{K}_c}(\mathbb{R}^p)}\right]. \tag{85}$$

As per Jensen's inequality

$$\left(\mathop{\mathbb{E}}_{\sigma}\left[\left\|\sum_{i=1}^{N}\sigma_i\mathcal{K}_c(\cdot,y^i)\right\|_{\mathcal{H}_{\mathcal{K}_c}(\mathbb{R}^p)}\right]\right)^2 \leq \mathop{\mathbb{E}}_{\sigma}\left[\left\|\sum_{i=1}^{N}\sigma_i\mathcal{K}_c(\cdot,y^i)\right\|^2_{\mathcal{H}_{\mathcal{K}_c}(\mathbb{R}^p)}\right], \tag{86}$$

and thus

$$\widehat{\mathcal{R}}_{\mathcal{D}}(\mathcal{M}_{\mathcal{D},c}) \leq \frac{1}{N}\sqrt{\mathop{\mathbb{E}}_{\sigma}\left[\left\|\sum_{i=1}^{N}\sigma_i\mathcal{K}_c(\cdot,y^i)\right\|^2_{\mathcal{H}_{\mathcal{K}_c}(\mathbb{R}^p)}\right]} \tag{87}$$

$$= \frac{1}{N}\sqrt{\mathop{\mathbb{E}}_{\sigma}\left[\left\langle\sum_{i=1}^{N}\sigma_i\mathcal{K}_c(\cdot,y^i),\sum_{i=1}^{N}\sigma_i\mathcal{K}_c(\cdot,y^i)\right\rangle_{\mathcal{H}_{\mathcal{K}_c}(\mathbb{R}^p)}\right]} \tag{88}$$

$$= \frac{1}{N}\sqrt{\mathop{\mathbb{E}}_{\sigma}\left[\sum_{i,j=1}^{N}\sigma_i\sigma_j\mathcal{K}_c(y^i,y^j)\right]} \tag{89}$$

$$= \frac{1}{N}\sqrt{\sum_{i,j=1}^{N}\mathcal{K}_c(y^i,y^j)\mathop{\mathbb{E}}_{\sigma}\left[\sigma_i\sigma_j\right]}. \tag{90}$$

Since $\sigma_1,\cdots,\sigma_N$ are independent random variables drawn from the Rademacher distribution, we have

$$\widehat{\mathcal{R}}_{\mathcal{D}}(\mathcal{M}_{\mathcal{D},c}) \leq \frac{1}{N}\sqrt{\sum_{i=1}^{N}\mathcal{K}_c(y^i,y^i)} \leq \frac{1}{\sqrt{N}}. \tag{91}$$

Hence, (30) and (31) follow.

D Proof of Lemma 3.2

The empirical Rademacher complexity of $\mathcal{L}_{\mathcal{D},c}$ is given as

$$\widehat{\mathcal{R}}_{\mathcal{D}}(\mathcal{L}_{\mathcal{D},c}) = \frac{1}{N}\mathop{\mathbb{E}}_{\sigma}\left[\sup_{l_{f_{y\mapsto z_c}}\in\mathcal{L}_{\mathcal{D},c}}\sum_{i=1}^{N}\sigma_i l_{f_{y\mapsto z_c}}(y^i,(z^i)_c)\right] \tag{92}$$

$$= \frac{1}{N}\mathop{\mathbb{E}}_{\sigma}\left[\sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}}\sum_{i=1}^{N}\sigma_i|(z^i)_c - f_{y\mapsto z_c}(y^i)|^2\right]. \tag{93}$$

Define

$$u_j(f_{y\mapsto z_c}) := \sum_{i=1}^{j}\sigma_i|(z^i)_c - f_{y\mapsto z_c}(y^i)|^2 \tag{94}$$

to express

$$\widehat{\mathcal{R}}_{\mathcal{D}}(\mathcal{L}_{\mathcal{D},c}) = \frac{1}{N}\mathop{\mathbb{E}}_{\sigma_1,\cdots,\sigma_{N-1}}\left[\mathop{\mathbb{E}}_{\sigma_N}\left[\sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}}\left(u_{N-1}(f_{y\mapsto z_c}) + \sigma_N|(z^N)_c - f_{y\mapsto z_c}(y^N)|^2\right)\right]\right]. \tag{95}$$

For any $\epsilon > 0$, let $f_1, f_2 \in \mathcal{M}_{\mathcal{D},c}$ be such that

$$\sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}} \left(u_{N-1}(f_{y\mapsto z_c}) + |(z^N)_c - f_{y\mapsto z_c}(y^N)|^2\right) = u_{N-1}(f_1) + |(z^N)_c - f_1(y^N)|^2 + \epsilon \tag{96}$$

$$\sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}} \left(u_{N-1}(f_{y\mapsto z_c}) - |(z^N)_c - f_{y\mapsto z_c}(y^N)|^2\right) = u_{N-1}(f_2) - |(z^N)_c - f_2(y^N)|^2 + \epsilon. \tag{97}$$

Consider,

$$\begin{aligned}
&\mathop{\mathbb{E}}_{\sigma_N}\left[\sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}} \left(u_{N-1}(f_{y\mapsto z_c}) + \sigma_N|(z^N)_c - f_{y\mapsto z_c}(y^N)|^2\right)\right] \\
&= \frac{1}{2}\sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}} \left(u_{N-1}(f_{y\mapsto z_c}) + |(z^N)_c - f_{y\mapsto z_c}(y^N)|^2\right) &(98)\\
&\quad + \frac{1}{2}\sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}} \left(u_{N-1}(f_{y\mapsto z_c}) - |(z^N)_c - f_{y\mapsto z_c}(y^N)|^2\right). &(99)
\end{aligned}$$

Thus,

$$\begin{aligned}
&\mathop{\mathbb{E}}_{\sigma_N}\left[\sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}} \left(u_{N-1}(f_{y\mapsto z_c}) + \sigma_N|(z^N)_c - f_{y\mapsto z_c}(y^N)|^2\right)\right] &(100)\\
&= \frac{1}{2}\left[u_{N-1}(f_1) + u_{N-1}(f_2)\right] + \frac{1}{2}\left[|(z^N)_c - f_1(y^N)|^2 - |(z^N)_c - f_2(y^N)|^2\right] + \epsilon &(101)\\
&= \frac{1}{2}\left[u_{N-1}(f_1) + u_{N-1}(f_2)\right] + \frac{1}{2}\left[\left(f_1(y^N) + f_2(y^N) - 2(z^N)_c\right)\left(f_1(y^N) - f_2(y^N)\right)\right] + \epsilon. &(102)
\end{aligned}$$

Define

$$\eta = \operatorname{sign}(f_1(y^N) - f_2(y^N)). \tag{103}$$

Since $f_1, f_2 \in \mathcal{M}_{\mathcal{D},c}$, we have $|f_1(y^N)| < 1$, $|f_2(y^N)| < 1$, and also $(z^N)_c \in \{0, 1\}$, leading to

$$f_1(y^N) + f_2(y^N) - 2(z^N)_c \leq 2, \tag{104}$$

so that

$$\begin{aligned}
&\mathop{\mathbb{E}}_{\sigma_N}\left[\sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}} \left(u_{N-1}(f_{y\mapsto z_c}) + \sigma_N|(z^N)_c - f_{y\mapsto z_c}(y^N)|^2\right)\right] \\
&\leq \frac{1}{2}\left[u_{N-1}(f_1) + u_{N-1}(f_2)\right] + \frac{1}{2}\left[2\eta\left(f_1(y^N) - f_2(y^N)\right)\right] + \epsilon &(105)\\
&= \frac{1}{2}\left[u_{N-1}(f_1) + 2\eta f_1(y^N)\right] + \frac{1}{2}\left[u_{N-1}(f_2) - 2\eta f_2(y^N)\right] + \epsilon &(106)\\
&\leq \frac{1}{2}\sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}} \left(u_{N-1}(f_{y\mapsto z_c}) + 2\eta f_{y\mapsto z_c}(y^N)\right) \\
&\quad + \frac{1}{2}\sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}} \left(u_{N-1}(f_{y\mapsto z_c}) - 2\eta f_{y\mapsto z_c}(y^N)\right) + \epsilon &(107)\\
&= \mathop{\mathbb{E}}_{\sigma_N}\left[\sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}} \left(u_{N-1}(f_{y\mapsto z_c}) + \sigma_N 2 f_{y\mapsto z_c}(y^N)\right)\right] + \epsilon. &(108)
\end{aligned}$$

Since the inequality holds for all $\epsilon > 0$, we have

$$\mathop{\mathbb{E}}_{\sigma_N}\left[\sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}}\left(u_{N-1}(f_{y\mapsto z_c})+\sigma_N|(z^N)_c-f_{y\mapsto z_c}(y^N)|^2\right)\right]$$
$$\leq \mathop{\mathbb{E}}_{\sigma_N}\left[\sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}}\left(u_{N-1}(f_{y\mapsto z_c})+\sigma_N 2 f_{y\mapsto z_c}(y^N)\right)\right]. \tag{109}$$

In other words,

$$\mathop{\mathbb{E}}_{\sigma_N}\left[\sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}}\sum_{i=1}^{N}\sigma_i|(z^i)_c-f_{y\mapsto z_c}(y^i)|^2\right]$$
$$\leq \mathop{\mathbb{E}}_{\sigma_N}\left[\sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}}\left(u_{N-1}(f_{y\mapsto z_c})+\sigma_N 2 f_{y\mapsto z_c}(y^N)\right)\right]. \tag{110}$$

That is,

$$\mathop{\mathbb{E}}_{\sigma_{N-1},\sigma_N}\left[\sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}}\sum_{i=1}^{N}\sigma_i|(z^i)_c-f_{y\mapsto z_c}(y^i)|^2\right]$$
$$\leq \mathop{\mathbb{E}}_{\sigma_N}\left[\mathop{\mathbb{E}}_{\sigma_{N-1}}\left[\sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}}\left(u_{N-2}(f_{y\mapsto z_c})+\sigma_N 2 f_{y\mapsto z_c}(y^N)+\sigma_{N-1}|z^{N-1}-f_{y\mapsto z_c}(y^{N-1})|^2\right)\right]\right] \tag{111}$$

We can follow the same procedure for σ_{N-1}, as did for σ_N for deriving (109), to show that

$$\mathop{\mathbb{E}}_{\sigma_{N-1}}\left[\sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}}\left(u_{N-2}(f_{y\mapsto z_c})+\sigma_N 2 f_{y\mapsto z_c}(y^N)+\sigma_{N-1}|z^{N-1}-f_{y\mapsto z_c}(y^{N-1})|^2\right)\right]$$
$$\leq \mathop{\mathbb{E}}_{\sigma_{N-1}}\left[\sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}}\left(u_{N-2}(f_{y\mapsto z_c})+\sigma_N 2 f_{y\mapsto z_c}(y^N)+\sigma_{N-1} 2 f_{y\mapsto z_c}(y^{N-1})\right)\right]. \tag{112}$$

That is,

$$\mathop{\mathbb{E}}_{\sigma_{N-1},\sigma_N}\left[\sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}}\sum_{i=1}^{N}\sigma_i|(z^i)_c-f_{y\mapsto z_c}(y^i)|^2\right]$$
$$\leq \mathop{\mathbb{E}}_{\sigma_{N-1},\sigma_N}\left[\sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}}\left(u_{N-2}(f_{y\mapsto z_c})+\sigma_{N-1} 2 f_{y\mapsto z_c}(y^{N-1})+\sigma_N 2 f_{y\mapsto z_c}(y^N)\right)\right]. \tag{113}$$

Proceeding in the same way, we get

$$\mathop{\mathbb{E}}_{\sigma_1,\cdots,\sigma_{N-1},\sigma_N}\left[\sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}}\sum_{i=1}^{N}\sigma_i|(z^i)_c-f_{y\mapsto z_c}(y^i)|^2\right]$$
$$\leq \mathop{\mathbb{E}}_{\sigma_1,\cdots,\sigma_{N-1},\sigma_N}\left[\sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}}\left(2\sum_{i=1}^{N}\sigma_i f_{y\mapsto z_c}(y^i)\right)\right] \tag{114}$$
$$= 2N\widehat{\mathcal{R}}_{\mathcal{D}}(\mathcal{M}_{\mathcal{D},c}). \tag{115}$$

Hence,

$$\widehat{\mathcal{R}}_{\mathcal{D}}(\mathcal{L}_{\mathcal{D},c}) \leq 2\widehat{\mathcal{R}}_{\mathcal{D}}(\mathcal{M}_{\mathcal{D},c}). \tag{116}$$

Now, using Theorem 3.1 leads to (35) and (36).

E Proof of Theorem 3.3

We define a function assessing the supremum of difference of expected loss value from the empirical averaged loss value:

$$\phi(\mathcal{D}) = \sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}} \left(\mathop{\mathbb{E}}_{(y,z)\sim\mathbb{P}_{y,z}} [l_{f_{y\mapsto z_c}}(y,z_c)] - \widehat{\mathbb{E}}_{\mathcal{D}}[l_{f_{y\mapsto z_c}}] \right). \tag{117}$$

Let $\mathcal{D}' = \{(y^1,z^1,q^1),\cdots,(y^{i-1},z^{i-1},q^{i-1}),(\tilde{y}^i,\tilde{z}^i,\tilde{q}^i),(y^{i+1},z^{i+1},q^{i+1}),\cdots,(y^N,z^N,q^N)\}$ be a "neighboring" set of $\mathcal{D}$ such that $\mathcal{D}'$ differs from $\mathcal{D}$ by only single entry, i.e., $(\tilde{y}^i,\tilde{z}^i,\tilde{q}^i)\notin\mathcal{D}$ and $(y^i,z^i,q^i)\notin\mathcal{D}'$. As the difference of suprema can't exceed the supremum of the difference, we have

$$\begin{aligned}
\phi(\mathcal{D}') - \phi(\mathcal{D}) \leq \sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}} \Bigg(& \mathop{\mathbb{E}}_{(y,z)\sim\mathbb{P}_{y,z}} [l_{f_{y\mapsto z_c}}(y,z_c)] - \widehat{\mathbb{E}}_{\mathcal{D}'}[l_{f_{y\mapsto z_c}}] \\
& - \mathop{\mathbb{E}}_{(y,z)\sim\mathbb{P}_{y,z}} [l_{f_{y\mapsto z_c}}(y,z_c)] + \widehat{\mathbb{E}}_{\mathcal{D}}[l_{f_{y\mapsto z_c}}] \Bigg) && (118)\\
= \sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}} & \left(\widehat{\mathbb{E}}_{\mathcal{D}}[l_{f_{y\mapsto z_c}}] - \widehat{\mathbb{E}}_{\mathcal{D}'}[l_{f_{y\mapsto z_c}}] \right) && (119)\\
= \sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}} & \frac{l_{f_{y\mapsto z_c}}(y^i,(z^i)_c) - l_{f_{y\mapsto z_c}}(\tilde{y}^i,(\tilde{z}^i)_c)}{N} && (120)\\
= \sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}} & \frac{|(z^i)_c - f_{y\mapsto z_c}(y^i)|^2 - |(\tilde{z}^i)_c - f_{y\mapsto z_c}(\tilde{y}^i)|^2}{N} && (121)\\
\leq & \frac{1}{N}, && (122)
\end{aligned}$$

where (122) follows from the facts that $(z^i)_c \in \{0,1\}$ and $f_{y\mapsto z_c}(y^i)\in[0,1]$. Similarly, we can obtain

$$\phi(\mathcal{D}) - \phi(\mathcal{D}') \leq \frac{1}{N}. \tag{123}$$

Thus

$$|\phi(\mathcal{D}) - \phi(\mathcal{D}')| \leq \frac{1}{N}. \tag{124}$$

Thus, ϕ satisfies the bounded differences property with bound $1/N$, and therefore by McDiarmid's inequality, for any $\epsilon > 0$, with probability at most $\exp(-2N\epsilon^2)$, the following holds:

$$\phi(\mathcal{D}) - \mathop{\mathbb{E}}_{\mathcal{D}\sim(\mathbb{P}_{y,z,q})^N} [\phi(\mathcal{D})] \geq \epsilon. \tag{125}$$

That is, with probability at most $\delta > 0$, the following holds:

$$\phi(\mathcal{D}) - \mathop{\mathbb{E}}_{\mathcal{D}\sim(\mathbb{P}_{y,z,q})^N} [\phi(\mathcal{D})] \geq \sqrt{\frac{\log(1/\delta)}{2N}}. \tag{126}$$

In other words, with probability at least $1-\delta$, the following holds:

$$\phi(\mathcal{D}) \leq \mathop{\mathbb{E}}_{\mathcal{D}\sim(\mathbb{P}_{y,z,q})^N} [\phi(\mathcal{D})] + \sqrt{\frac{\log(1/\delta)}{2N}}. \tag{127}$$

Let $\tilde{\mathcal{D}} = \left\{(\tilde{y}^i, \tilde{z}^i, \tilde{q}^i) \mid i \in \{1, 2, \cdots, N\}\right\} \sim (\mathbb{P}_{y,z,q})^N$ be another set of i.i.d. samples. Consider

$$\mathop{\mathbb{E}}_{\mathcal{D}\sim(\mathbb{P}_{y,z,q})^N}[\phi(\mathcal{D})] = \mathop{\mathbb{E}}_{\mathcal{D}\sim(\mathbb{P}_{y,z,q})^N}\left[\sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}}\left(\mathop{\mathbb{E}}_{(y,z)\sim\mathbb{P}_{y,z}}[l_{f_{y\mapsto z_c}}(y,z_c)] - \widehat{\mathbb{E}}_{\mathcal{D}}[l_{f_{y\mapsto z_c}}]\right)\right] \tag{128}$$

$$= \mathop{\mathbb{E}}_{\mathcal{D}\sim(\mathbb{P}_{y,z,q})^N}\left[\sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}}\left(\mathop{\mathbb{E}}_{\tilde{\mathcal{D}}\sim(\mathbb{P}_{y,z,q})^N}\left[\widehat{\mathbb{E}}_{\tilde{\mathcal{D}}}[l_{f_{y\mapsto z_c}}] - \widehat{\mathbb{E}}_{\mathcal{D}}[l_{f_{y\mapsto z_c}}]\right]\right)\right], \tag{129}$$

where we have used the fact that

$$\mathop{\mathbb{E}}_{\tilde{\mathcal{D}}\sim(\mathbb{P}_{y,z,q})^N}\left[\widehat{\mathbb{E}}_{\tilde{\mathcal{D}}}[l_{f_{y\mapsto z_c}}]\right] = \mathop{\mathbb{E}}_{(y,z,q)\sim\mathbb{P}_{y,z,q}}[l_{f_{y\mapsto z_c}}(y,z_c)] = \mathop{\mathbb{E}}_{(y,z)\sim\mathbb{P}_{y,z}}[l_{f_{y\mapsto z_c}}(y,z_c)]. \tag{130}$$

It follows from (129) that

$$\begin{aligned}
&\mathop{\mathbb{E}}_{\mathcal{D}\sim(\mathbb{P}_{y,z,q})^N}[\phi(\mathcal{D})] \\
&\leq \mathop{\mathbb{E}}_{\mathcal{D}\sim(\mathbb{P}_{y,z,q})^N,\tilde{\mathcal{D}}\sim(\mathbb{P}_{y,z,q})^N}\left[\sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}}\left(\widehat{\mathbb{E}}_{\tilde{\mathcal{D}}}[l_{f_{y\mapsto z_c}}] - \widehat{\mathbb{E}}_{\mathcal{D}}[l_{f_{y\mapsto z_c}}]\right)\right] &(131)\\
&= \mathop{\mathbb{E}}_{\mathcal{D}\sim(\mathbb{P}_{y,z,q})^N,\tilde{\mathcal{D}}\sim(\mathbb{P}_{y,z,q})^N}\left[\sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}}\left(\frac{1}{N}\sum_{i=1}^{N}\left(l_{f_{y\mapsto z_c}}\left(\tilde{y}^i,(\tilde{z}^i)_c\right) - l_{f_{y\mapsto z_c}}\left(y^i,(z^i)_c\right)\right)\right)\right] &(132)\\
&= \mathop{\mathbb{E}}_{\mathcal{D}\sim(\mathbb{P}_{y,z,q})^N,\tilde{\mathcal{D}}\sim(\mathbb{P}_{y,z,q})^N,\sigma}\left[\sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}}\left(\frac{1}{N}\sum_{i=1}^{N}\sigma_i\left(l_{f_{y\mapsto z_c}}\left(\tilde{y}^i,(\tilde{z}^i)_c\right) - l_{f_{y\mapsto z_c}}\left(y^i,(z^i)_c\right)\right)\right)\right], &(133)
\end{aligned}$$

where we have considered the facts that

- $\sigma_1, \cdots, \sigma_N$ are Rademacher variables (i.e. taking values in $\{-1, 1\}$ with probability equal to 1/2), and
- we have

$$\begin{aligned}
&\mathop{\mathbb{E}}_{\mathcal{D}\sim(\mathbb{P}_{y,z,q})^N,\tilde{\mathcal{D}}\sim(\mathbb{P}_{y,z,q})^N}\left[\sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}}\left(\frac{1}{N}\sum_{i=1}^{N}\left(l_{f_{y\mapsto z_c}}\left(\tilde{y}^i,(\tilde{z}^i)_c\right) - l_{f_{y\mapsto z_c}}\left(y^i,(z^i)_c\right)\right)\right)\right] \\
&= \mathop{\mathbb{E}}_{\mathcal{D}\sim(\mathbb{P}_{y,z,q})^N,\tilde{\mathcal{D}}\sim(\mathbb{P}_{y,z,q})^N}\left[\sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}}\left(\frac{1}{N}\sum_{i=1}^{N}\left(l_{f_{y\mapsto z_c}}(y^i,(z^i)_c) - l_{f_{y\mapsto z_c}}(\tilde{y}^i,(\tilde{z}^i)_c)\right)\right)\right]. &(134)
\end{aligned}$$

It follows from (133) that

$$\begin{aligned}
\mathop{\mathbb{E}}_{\mathcal{D}\sim(\mathbb{P}_{y,z,q})^N}[\phi(\mathcal{D})] \leq &\mathop{\mathbb{E}}_{\tilde{\mathcal{D}}\sim(\mathbb{P}_{y,z,q})^N,\sigma}\left[\sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}}\left(\frac{1}{N}\sum_{i=1}^{N}\sigma_i l_{f_{y\mapsto z_c}}\left(\tilde{y}^i,(\tilde{z}^i)_c\right)\right)\right] \\
&+ \mathop{\mathbb{E}}_{\mathcal{D}\sim(\mathbb{P}_{y,z,q})^N,\sigma}\left[\sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}}\left(\frac{1}{N}\sum_{i=1}^{N}-\sigma_i l_{f_{y\mapsto z_c}}\left(y^i,(z^i)_c\right)\right)\right]. &(135)
\end{aligned}$$

Since σ_i and $-\sigma_i$ are distributed identically, we have

$$\mathop{\mathbb{E}}_{\mathcal{D}\sim(\mathbb{P}_{y,z,q})^N}[\phi(\mathcal{D})] \leq 2 \mathop{\mathbb{E}}_{\mathcal{D}\sim(\mathbb{P}_{y,z,q})^N,\sigma}\left[\sup_{f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}}\left(\frac{1}{N}\sum_{i=1}^{N}\sigma_i l_{f_{y\mapsto z_c}}\left(y^i,(z^i)_c\right)\right)\right] \tag{136}$$

$$= 2 \mathop{\mathbb{E}}_{\mathcal{D}\sim(\mathbb{P}_{y,z,q})^N,\sigma}\left[\sup_{l_{f_{y\mapsto z_c}}\in\mathcal{L}_{\mathcal{D},c}}\left(\frac{1}{N}\sum_{i=1}^{N}\sigma_i l_{f_{y\mapsto z_c}}\left(y^i,(z^i)_c\right)\right)\right] \tag{137}$$

$$= 2 \mathop{\mathbb{E}}_{\mathcal{D}\sim(\mathbb{P}_{y,z,q})^N}\left[\widehat{\mathcal{R}}_{\mathcal{D}}(\mathcal{L}_{\mathcal{D},c})\right] \tag{138}$$

$$\leq \frac{4}{\sqrt{N}}, \tag{139}$$

where we have used Lemma 3.2. Thus, with probability at least $1-\delta$, we have

$$\phi(\mathcal{D}) \leq \frac{4}{\sqrt{N}} + \sqrt{\frac{\log(1/\delta)}{2N}}. \tag{140}$$

Therefore, for any $f_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}$, we have with probability at least $1-\delta$,

$$\mathop{\mathbb{E}}_{(y,z)\sim\mathbb{P}_{y,z}}[l_{f_{y\mapsto z_c}}(y,z_c)] \leq \frac{1}{N}\sum_{i=1}^{N} l_{f_{y\mapsto z_c}}(y^i,(z^i)_c) + \frac{4}{\sqrt{N}} + \sqrt{\frac{\log(1/\delta)}{2N}}. \tag{141}$$

F Proof of Theorem 4.2

Since $f^*_{y\mapsto z_c}\in\mathcal{M}_{\mathcal{D},c}$, it follows from Theorem 3.3 that we have with probability at least $1-\delta$ for any $\delta\in(0,1)$,

$$\mathop{\mathbb{E}}_{(y,z)\sim\mathbb{P}_{y,z}}\left[\left|z_c - f^*_{y\mapsto z_c}(y)\right|^2\right] \leq \frac{1}{N}\sum_{i=1}^{N}\left|(z^i)_c - f^*_{y\mapsto z_c}(y^i)\right|^2 + \frac{4}{\sqrt{N}} + \sqrt{\frac{\log(1/\delta)}{2N}}. \tag{142}$$

Due to Assumption 1 and Assumption 2,

$$f^*_{y\mapsto z_c}(y^i) \approx (z^i)_c,\ \forall i\in\{1,2,\cdots,N\}. \tag{143}$$

That is,

$$\frac{1}{N}\sum_{i=1}^{N}\left|(z^i)_c - f^*_{y\mapsto z_c}(y^i)\right|^2 \approx 0. \tag{144}$$

Hence, the result follows.

G Proof of Theorem 4.3

Consider

$$\mathop{\mathbb{E}}_{(y,z)\sim\mathbb{P}_{y,z}}\left[\left|z_c - f^*_{y\mapsto z_c}(y)\right|^2\right] - \mathop{\mathbb{E}}_{(y,z)\sim\mathbb{P}_{y,z}}\left[\left|z_c - \mathbb{P}_{z|y}(z_c=1|y)\right|^2\right]$$

$$= \mathop{\mathbb{E}}_{(y,z)\sim\mathbb{P}_{y,z}}\left[|f^*_{y\mapsto z_c}(y)|^2 - |\mathbb{P}_{z|y}(z_c=1|y)|^2 - 2z_c\left(f^*_{y\mapsto z_c}(y) - \mathbb{P}_{z|y}(z_c=1|y)\right)\right] \tag{145}$$

$$= \mathop{\mathbb{E}}_{y\sim\mathbb{P}_y}\left[|f^*_{y\mapsto z_c}(y)|^2\right] - \mathop{\mathbb{E}}_{y\sim\mathbb{P}_y}\left[|\mathbb{P}_{z|y}(z_c=1|y)|^2\right]$$
$$-2\mathop{\mathbb{E}}_{y\sim\mathbb{P}_y}\left[\mathbb{P}_{z|y}(z_c=1|y)\left(f^*_{y\mapsto z_c}(y) - \mathbb{P}_{z|y}(z_c=1|y)\right)\right] \tag{146}$$

$$= \mathop{\mathbb{E}}_{y\sim\mathbb{P}_y}\left[|f^*_{y\mapsto z_c}(y) - \mathbb{P}_{z|y}(z_c=1|y)|^2\right]. \tag{147}$$

Thus,

$$\mathop{\mathbb{E}}_{y\sim\mathbb{P}_y}\left[|f^*_{y\mapsto z_c}(y)-\mathbb{P}_{z|y}(z_c=1|y)|^2\right]\leq \mathop{\mathbb{E}}_{(y,z)\sim\mathbb{P}_{y,z}}\left[\left|z_c-f^*_{y\mapsto z_c}(y)\right|^2\right]. \tag{148}$$

Hence, the results follow by using (148) in Theorem 3.3 and Theorem 4.2.

H Proof of Theorem 4.6

It follows from (18) that

$$\Gamma_{\mathcal{G}_c}(y)=\Gamma_{\mathcal{A}_{\Upsilon^{c,q^*(y)}}}(y). \tag{149}$$

Using (14), we get

$$\Gamma_{\mathcal{G}_c}(y)<\left(1+\frac{p\left|\mathcal{I}^{c,q^*(y)}\right|^2}{2\left\|\Upsilon^{c,q^*(y)}\right\|_F^2}\right)\left\|\left[\begin{array}{ccc} y-y^{\Gamma_1^{c,q^*(y)}} & \cdots & y-y^{\Gamma^{c,q^*(y)}_{|\mathcal{I}^{c,q^*(y)}|}}\end{array}\right]\right\|_2. \tag{150}$$

Using (150) in (44), we get (60).

Received 5 July 2024; revised 14 May 2025; accepted 18 May 2025