Data Collaboration Analysis with Orthonormal Basis Selection and Alignment

Keiyu Nosaka[a], Yamato Suetake[b], Yuichi Takano[c], Akiko Yoshise[c,*]

[a]*Graduate School of Science and Technology, University of Tsukuba, 1-1-1, Tennodai, Tsukuba, 305-8573, Ibaraki, Japan*
[b]*School of Science and Engineering, University of Tsukuba, 1-1-1, Tennodai, Tsukuba, 305-8573, Ibaraki, Japan*
[c]*Institute of Systems and Information Engineering, and the Center for Artificial Intelligence Research, Tsukuba Institute for Advanced Research (TIAR), University of Tsukuba, 1-1-1, Tennodai, Tsukuba, 305-8573, Ibaraki, Japan*

Abstract

Data Collaboration (DC) enables multiple parties to jointly train a model by sharing only linear projections of their private datasets. The core challenge in DC is to align the bases of these projections without revealing each party's secret basis. While existing theory suggests that any target basis spanning the common subspace should suffice, in practice, the choice of basis can substantially affect both accuracy and numerical stability. We introduce **Orthonormal Data Collaboration (ODC)**, which enforces orthonormal secret and target bases, thereby reducing alignment to the classical Orthogonal Procrustes problem, which admits a closed-form solution. We prove that the resulting change-of-basis matrices achieve *orthogonal concordance*, aligning all parties' representations up to a shared orthogonal transform and rendering downstream performance invariant to the target basis. Computationally, ODC reduces the alignment complexity from $O\big(\min\{a(c\ell)^2,\ a^2c\ell\}\big)$ to $O(ac\ell^2)$, and empirical evaluations show up to 100× speed-ups with equal or better accuracy across benchmarks. ODC preserves DC's one-round communication pattern and privacy assumptions, providing a simple and efficient drop-in improvement to existing DC pipelines.

Keywords: Data Collaboration Analysis, Orthogonal Procrustes Problem, Privacy-Preserving Machine Learning

1. Introduction

The effectiveness of machine learning (ML) algorithms depends strongly on the quality, diversity, and comprehensiveness of the training datasets. High-quality datasets enhance predictive performance and improve the generalization capabilities of ML models across diverse real-world applications. To overcome the biases and limitations inherent to datasets from a single source, data aggregation across multiple sources has become a common practice. Nonetheless, this practice introduces significant ethical and privacy concerns, particularly regarding unauthorized access and disclosure of sensitive user information. Recent literature highlights a growing awareness and reports an increasing number of privacy breaches linked to large-scale personal data collection and analysis [1].

To address these privacy concerns, Privacy-Preserving Machine Learning (PPML) has emerged as a crucial approach, enabling the secure and effective use of sensitive data—such as medical records, financial transactions, and geolocation data—without compromising individual privacy. Among various PPML methodologies, *Federated Learning* (FL) [2] has gained prominence for enabling collaborative model training among decentralized data sources, safeguarding data privacy by limiting direct data exchange.

While FL offers promising solutions, it faces notable challenges in *cross-silo* settings, primarily due to its reliance on iterative communication among participating entities during training. This communication overhead constitutes a significant obstacle, particularly in privacy-sensitive sectors such as healthcare and finance, where institutions often operate under stringent regulatory constraints and within isolated network infrastructures. Furthermore, FL inherently

⋆Earlier versions of this article have been circulated under the titles "Data Collaboration Analysis Over Matrix Manifolds" and "Data Collaboration Analysis with Orthogonal Basis Alignment."

*Corresponding author

Email addresses: s2430118@u.tsukuba.ac.jp (Keiyu Nosaka), s2211759@u.tsukuba.ac.jp (Yamato Suetake), ytakano@sk.tsukuba.ac.jp (Yuichi Takano), yoshise@sk.tsukuba.ac.jp (Akiko Yoshise)

lacks formal privacy guarantees, necessitating the use of supplementary privacy-enhancing mechanisms to achieve explicit privacy assurances.

A widely adopted strategy to mitigate these shortcomings is *Differential Privacy* (DP) [3, 4], which protects individual records by adding carefully calibrated noise to the learning process. Although DP provides strong formal privacy guarantees, it necessarily trades off model utility for privacy [5], and the resulting degradation in accuracy can substantially limit its practicality in real-world deployments, especially when data are already scarce or biased because they originate from a small number of institutions.

To overcome the limitations of conventional PPML methods, *Data Collaboration* (DC) has emerged as a promising alternative [6, 7]. Unlike FL, DC leverages the central aggregation of *secure intermediate representations* computed locally from raw data. Specifically, each participating entity independently transforms its dataset using a basis matrix that it has privately selected. Subsequently, a central aggregator aligns these transformed datasets within a common representation space by constructing a shared target basis—*without knowledge of the secret bases*. This approach facilitates collaborative model training while ensuring robust privacy preservation, thus eliminating the requirement for iterative communication among semi-honest parties [8].

Recent advancements in DC have expanded its applicability to scenarios involving more serious adversarial threats. Notably, methods have been proposed to counter re-identification attacks targeting intermediate representations [9], ensuring that transformed datasets remain unlinkable to their sources. Additionally, integrating differential privacy mechanisms has been explored to further mitigate the risks associated with malicious collusion among participants [10]. Moreover, hybrid approaches such as FedDCL [11] demonstrate how DC's advantages can be effectively combined with FL paradigms, resulting in scalable, privacy-preserving collaborative ML solutions.

Although DC has empirically demonstrated significant potential in balancing privacy and utility without iterative communication, its theoretical foundations remain underdeveloped. Existing theoretical analyses typically assume that *any* target basis spanning the same subspace as the secret bases is sufficient. However, recent empirical studies indicate that selecting the target basis substantially influences the performance of downstream models [12, 13]. Specifically, choosing target bases that disproportionately emphasize certain directions in the feature space can degrade model accuracy and utility. These findings highlight a clear discrepancy between current theoretical guarantees and observed empirical behavior, emphasizing the need for improved basis selection and alignment strategies within the DC framework.

1.1. Our Contributions

We propose a novel framework, termed *Orthonormal Data Collaboration (ODC)*, to bridge the gap between DC's theoretical foundations and empirical performance. The central innovation of ODC is the explicit enforcement of *orthonormality* constraints on both the secret and target bases during basis selection and alignment. This design leverages common practices in DC, as conventional dimensionality-reduction methods, such as Principal Component Analysis (PCA) and Singular Value Decomposition (SVD), naturally produce orthonormal bases, thereby imposing minimal additional overhead.

The orthonormality constraint leads to two significant theoretical advantages:

1. **Alignment Efficiency:** The basis alignment simplifies precisely to the classical *Orthogonal Procrustes Problem* [14], for which a closed-form analytical solution is available. This simplification significantly reduces the computational complexity relative to existing DC approaches.

2. **Orthogonal Concordance:** All orthonormal target bases spanning the same subspace as the secret bases result in identical downstream model performance. Thus, the specific choice of an orthonormal target basis becomes inconsequential, effectively resolving previous instabilities in DC.

Empirical evaluations confirm these theoretical results by: (a) demonstrating that ODC achieves substantially faster alignment compared to state-of-the-art DC methods, validating its theoretical complexity advantages; (b) highlighting the practical benefits of orthogonal concordance in stabilizing and enhancing model accuracy; and (c) assessing the robustness of ODC under realistic conditions where theoretical assumptions may not strictly hold.

Notably, the ODC framework extends traditional DC by adding only a single practical assumption—the orthonormality of secret bases. Empirical results demonstrate that relaxing this assumption notably degrades performance,

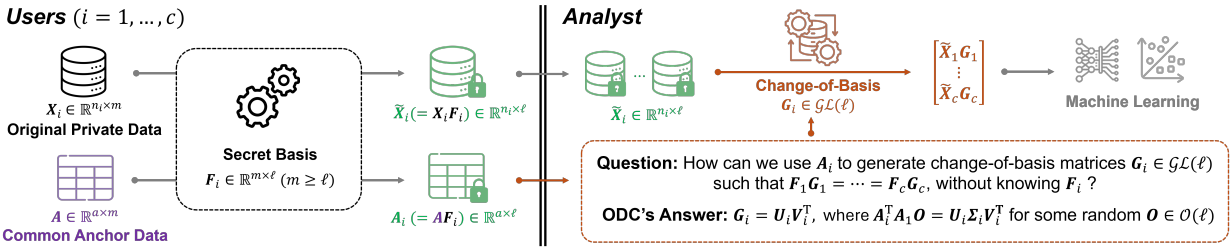


Figure 1: Conceptual illustration of the Orthonormal Data Collaboration (ODC) framework. Each participating user independently projects their private dataset $\boldsymbol{X}_i \in \mathbb{R}^{n_i \times m}$ and a common anchor dataset $\boldsymbol{A} \in \mathbb{R}^{a \times m}$ into intermediate representations $\tilde{\boldsymbol{X}}_i = \boldsymbol{X}_i \boldsymbol{F}_i$ and $\boldsymbol{A}_i = \boldsymbol{A}\boldsymbol{F}_i$, respectively, using a privately selected orthonormal secret basis $\boldsymbol{F}_i \in \mathbb{R}^{m \times \ell}$. To collaboratively train machine learning models without revealing their private raw data, an analyst constructs orthogonal change-of-basis matrices $\boldsymbol{G}_i \in O(\ell) := \{\boldsymbol{O} \in \mathbb{R}^{\ell \times \ell} : \boldsymbol{O}^\top \boldsymbol{O} = \boldsymbol{O}\boldsymbol{O}^\top = \boldsymbol{I}\}$ to align these representations onto a shared orthonormal target basis, without directly accessing the private secret bases. The ODC framework ensures these matrices achieve *orthogonal concordance*, aligning all user representations up to a common orthogonal transformation. Consequently, the analyst can safely aggregate and analyze the aligned representations $\tilde{\boldsymbol{X}}_i \boldsymbol{G}_i$ to perform downstream machine learning tasks. ODC explicitly addresses a key practical question: "How can we use only the intermediate anchor representations $\boldsymbol{A}_i$ to generate alignment matrices $\boldsymbol{G}_i$ without explicitly knowing the secret bases $\boldsymbol{F}_i$?" The analytical solution, illustrated in the figure, is $\boldsymbol{G}_i = \boldsymbol{U}_i \boldsymbol{V}_i^\top$, computed via the singular value decomposition $\boldsymbol{A}_i^\top \boldsymbol{A}_1 \boldsymbol{O} = \boldsymbol{U}_i \boldsymbol{\Sigma}_i \boldsymbol{V}_i^\top$, where $\boldsymbol{O} \in O(\ell)$ is an arbitrarily selected orthogonal matrix.

underscoring its necessity. Because orthonormal bases already emerge naturally in existing DC workflows (e.g., [9], DP integration [10], and FL integration [11]), ODC can seamlessly integrate into current pipelines. Moreover, our empirical comparisons position ODC within the broader context of PPML, highlighting its advantages over mainstream techniques such as DP-based perturbations and FL.

Fig. 1 provides a conceptual illustration of ODC, highlighting the principle of orthogonal concordance achieved via basis alignment.

1.2. Notations and Organization

Notations. For a positive integer c, we write $[c] := \{1, 2, \ldots, c\}$. We denote by $\mathbb{R}^{p \times q}$ the set of real $p \times q$ matrices. For a matrix $\boldsymbol{M}$, $\boldsymbol{M}^\top$ denotes the transpose, $\boldsymbol{M}^\dagger$ the Moore–Penrose pseudoinverse [15] (e.g., $M^\dagger = (M^\top M)^{-1} M^\top$ when M has full column rank), $\|\boldsymbol{M}\|_F$ the Frobenius norm, and $\mathrm{tr}(\boldsymbol{M})$ the matrix trace. We write

$$\mathcal{GL}(\ell) := \{\boldsymbol{G} \in \mathbb{R}^{\ell \times \ell} : \mathrm{rank}(\boldsymbol{G}) = \ell\}, \qquad O(\ell) := \{\boldsymbol{O} \in \mathbb{R}^{\ell \times \ell} : \boldsymbol{O}^\top \boldsymbol{O} = \boldsymbol{O}\boldsymbol{O}^\top = \boldsymbol{I}_\ell\} \tag{1}$$

for the general linear and orthogonal groups, respectively. For a matrix $\boldsymbol{M}$, its (thin) SVD is written as $\boldsymbol{M} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top$.

In the DC setting, each user $i \in [c]$ holds a private dataset $\boldsymbol{X}_i \in \mathbb{R}^{n_i \times m}$ (with labels $\boldsymbol{L}_i$) and a common anchor dataset $\boldsymbol{A} \in \mathbb{R}^{a \times m}$. User i selects a secret basis $\boldsymbol{F}_i \in \mathbb{R}^{m \times \ell}$ and releases only the intermediate representations

$$\tilde{\boldsymbol{X}}_i := \boldsymbol{X}_i \boldsymbol{F}_i \in \mathbb{R}^{n_i \times \ell}, \qquad \boldsymbol{A}_i := \boldsymbol{A}\boldsymbol{F}_i \in \mathbb{R}^{a \times \ell}. \tag{2}$$

The analyst constructs change-of-basis matrices $\boldsymbol{G}_i \in \mathbb{R}^{\ell \times \ell}$ and forms aligned data

$$\hat{\boldsymbol{X}}_i := \tilde{\boldsymbol{X}}_i \boldsymbol{G}_i. \tag{3}$$

Contemporary DC methods typically allow $\boldsymbol{G}_i \in \mathcal{GL}(\ell)$, whereas ODC enforces $\boldsymbol{G}_i \in O(\ell)$.

Organization. § 2 reviews the DC protocol as well as its privacy and communication properties. § 3 discusses related work on Procrustes-based alignment and PPML. § 4 summarizes existing DC basis-alignment methods and their limitations. § 5 presents ODC, derives its closed-form Procrustes alignment, and proves orthogonal concordance. § 6–§ 9 report empirical studies on time efficiency, robustness under relaxed assumptions, concordance, and anchor construction. § 10 concludes the paper.

2. Preliminaries

In this section, we present the necessary preliminaries on DC analysis. Specifically, we begin with an overview of the DC algorithm, followed by an examination of its privacy-preserving mechanisms and communication overhead.

2.1. The Data Collaboration Algorithm

We consider a general DC framework for supervised machine learning [6, 10]. Let $\boldsymbol{X} \in \mathbb{R}^{n\times m}$ represent a dataset containing n training samples, each characterized by m features, and let $\boldsymbol{L} \in \mathbb{R}^{n\times \ell}$ denote the corresponding label set with ℓ labels. For privacy-preserving analysis across multiple entities, we assume the dataset is horizontally partitioned among c distinct entities, expressed as:

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{X}_1 \\ \vdots \\ \boldsymbol{X}_c \end{bmatrix}, \quad \boldsymbol{L} = \begin{bmatrix} \boldsymbol{L}_1 \\ \vdots \\ \boldsymbol{L}_c \end{bmatrix}, \tag{4}$$

where each entity i possesses a subset of the data $\boldsymbol{X}_i \in \mathbb{R}^{n_i\times m}$ and corresponding labels $\boldsymbol{L}_i \in \mathbb{R}^{n_i\times \ell}$. The total number of samples satisfies $n = \sum_{i\in[c]} n_i$, where $[c] := \{1, 2, \ldots, c\}$. Additionally, each entity holds a test dataset $\boldsymbol{Y}_i \in \mathbb{R}^{s_i\times m}$, for which the goal is to predict the corresponding labels $\boldsymbol{L}_{\boldsymbol{Y}_i} \in \mathbb{R}^{s_i\times \ell}$. The DC framework can also be extended to handle more complex scenarios, such as partially shared features [16] or data partitioned both horizontally and vertically [7].

The framework defines two primary roles: the *user* and the *analyst*. Users possess their private datasets $\boldsymbol{X}_i$ and corresponding labels $\boldsymbol{L}_i$, and aim to improve local model performance by leveraging insights from other users' data without revealing their own. The analyst's role is to facilitate this collaborative process by providing the necessary resources and infrastructure for the machine learning workflow.

At the outset, users collaboratively construct a shared anchor dataset $\boldsymbol{A} \in \mathbb{R}^{a\times m}$, where $a > m$. Following the standard DC literature [6, 17], we assume that $\boldsymbol{A}$ is either (i) synthetically generated dummy data or (ii) sampled from an external public database (for example, unlabeled compounds from PubChem for chemical fingerprints [17]). Here, "public" refers only to the *data source*: the concrete subset of records and feature vectors that form $\boldsymbol{A}$ is sampled locally by the users and is *not* disclosed to the analyst. The analyst receives only the transformed anchor representations $\boldsymbol{A}_i = \boldsymbol{A}\boldsymbol{F}_i$, not the raw anchor $\boldsymbol{A}$ itself, so $\boldsymbol{A}$ remains unknown to the analyst unless a malicious colluding user explicitly reveals it.

Each user independently selects an m-dimensional basis with size ℓ, denoted by $\boldsymbol{F}_i \in \mathbb{R}^{m\times \ell}$ ($m \geq \ell$), to linearly transform their private dataset $\boldsymbol{X}_i$ and the anchor dataset $\boldsymbol{A}$ into secure intermediate representations. A common basis selection method employs truncated SVD with a random orthogonal mapping [11, 17]:

$$\boldsymbol{F}_i = \boldsymbol{V}_i\boldsymbol{E}_i, \tag{5}$$

where, $\boldsymbol{E}_i \in O(\ell) := \{\boldsymbol{O} \in \mathbb{R}^{\ell\times\ell} : \boldsymbol{O}^\top\boldsymbol{O} = \boldsymbol{O}\boldsymbol{O}^\top = \boldsymbol{I}\}$ and $\boldsymbol{V}_i \in \mathbb{R}^{m\times\ell}$ denotes the top ℓ right singular vectors of $\boldsymbol{X}_i$. Notably, this typical method inherently produces orthonormal bases, i.e., $\boldsymbol{F}_i^\top\boldsymbol{F}_i = \boldsymbol{I}$.

Once the secret bases $\boldsymbol{F}_i$ are chosen for each user, the secure intermediate representations of the private dataset $\boldsymbol{X}_i$ and the anchor dataset $\boldsymbol{A}$ are computed as follows.

$$\tilde{\boldsymbol{X}}_i = \boldsymbol{X}_i\boldsymbol{F}_i, \quad \boldsymbol{A}_i = \boldsymbol{A}\boldsymbol{F}_i. \tag{6}$$

Each user shares $\tilde{\boldsymbol{X}}_i$, $\boldsymbol{L}_i$, and $\boldsymbol{A}_i$ with the analyst, whose task is to construct a collaborative ML model based on all $\tilde{\boldsymbol{X}}_i$ and $\boldsymbol{L}_i$. However, directly concatenating $\tilde{\boldsymbol{X}}_i$ and building a model from it is futile, as the bases were selected privately and are generally different. Within the DC framework, the analyst aims to align the secret bases using change-of-basis matrices $\boldsymbol{G}_i \in \mathbb{R}^{\ell\times\ell}$, and constructs $\hat{\boldsymbol{X}}$ as follows:

$$\hat{\boldsymbol{X}} = \begin{bmatrix} \tilde{\boldsymbol{X}}_1\boldsymbol{G}_1 \\ \vdots \\ \tilde{\boldsymbol{X}}_c\boldsymbol{G}_c \end{bmatrix}. \tag{7}$$

After successfully creating the change-of-basis matrices from the aggregated $\boldsymbol{A}_i$, as detailed in § 3 and 5, the analyst utilizes $\hat{\boldsymbol{X}}$ and $\boldsymbol{L}$ to construct a supervised classification model h:

$$\boldsymbol{L} \approx h(\hat{\boldsymbol{X}}). \tag{8}$$

This model h can simply be distributed to the users along with $\boldsymbol{G}_i$ to predict the labels $\boldsymbol{L}_{\boldsymbol{Y}_i}$ of the test dataset $\boldsymbol{Y}_i$:

$$\boldsymbol{L}_{\boldsymbol{Y}_i} = h(\boldsymbol{Y}_i\boldsymbol{F}_i\boldsymbol{G}_i), \tag{9}$$

Algorithm 1: Overview of the DC algorithm *(Adapted from Algorithm 1 in [9])*

Input : $\boldsymbol{X}_i \in \mathbb{R}^{n_i \times m}$, $\boldsymbol{L}_i \in \mathbb{R}^{n_i \times \ell}$, $\boldsymbol{Y}_i \in \mathbb{R}^{s_i \times m}$ for each user $i \in [c]$
Output: $\boldsymbol{L}_{\boldsymbol{Y}_i} \in \mathbb{R}^{s_i \times \ell}$ for each user $i \in [c]$

1: **User-side** ($i \in [c]$): **begin**
2: Generate $\boldsymbol{A} \in \mathbb{R}^{a \times m}$ and share it with all users
3: Select a secret basis $\boldsymbol{F}_i \in \mathbb{R}^{m \times \ell}$
4: Compute $\tilde{\boldsymbol{X}}_i = \boldsymbol{X}_i \boldsymbol{F}_i$ and $\boldsymbol{A}_i = \boldsymbol{A}\boldsymbol{F}_i$
5: Share $\tilde{\boldsymbol{X}}_i$, $\boldsymbol{A}_i$, and $\boldsymbol{L}_i$ with the analyst
6: **end**
7: **Analyst-side**: **begin**
8: **for** *each user* $i \in [c]$ **do**
9: Obtain $\tilde{\boldsymbol{X}}_i$, $\boldsymbol{A}_i$, and $\boldsymbol{L}_i$
10: Generate a change-of-basis matrix $\boldsymbol{G}_i \in \mathbb{R}^{\ell \times \ell}$
11: Compute $\hat{\boldsymbol{X}}_i = \tilde{\boldsymbol{X}}_i \boldsymbol{G}_i$
12: **end**
13: Set $\hat{\boldsymbol{X}}$ and $\boldsymbol{L}$ by aggregating all $\hat{\boldsymbol{X}}_i$ and $\boldsymbol{L}_i$
14: Analyze $\hat{\boldsymbol{X}}$ to obtain h such that $\boldsymbol{L} \approx h(\hat{\boldsymbol{X}})$
15: Return $\boldsymbol{G}_i$ and h to each user $i \in [c]$
16: **end**
17: **User-side** ($i \in [c]$): **begin**
18: Obtain $\boldsymbol{G}_i$ and h
19: Predict $\boldsymbol{L}_{\boldsymbol{Y}_i} = h(\boldsymbol{Y}_i \boldsymbol{F}_i \boldsymbol{G}_i)$
20: **end**

or employed in other DC-based applications [9, 18, 19, 20, 21, 22, 23] for enhanced privacy or utility. An overview of the DC algorithm is presented in **Algorithm** 1. Notably, cross-entity communication within this framework is strictly limited to Steps 2, 5, and 15 of **Algorithm** 1. The primary challenge of the DC algorithm lies in Step 10, which poses the key question: *How can we generate the change-of-basis matrices $\boldsymbol{G}_i$ without access to the secret bases $\boldsymbol{F}_i$?* This question is addressed in detail in § 4 and § 5.

2.2. Privacy Analysis

This section provides a brief review of the privacy guarantees and limitations inherent in the DC framework [8]. Throughout this paper, we adopt the standard *semi-honest* threat model for DC [8, 9, 10]: all parties follow the prescribed protocol but may attempt to infer additional information from the data they are authorized to observe.

Standard semi-honest model. Each user i observes only its own raw dataset $(\boldsymbol{X}_i, \boldsymbol{L}_i)$, the shared anchor $\boldsymbol{A}$, and the learned model h. The analyst observes only intermediate representations $(\tilde{\boldsymbol{X}}_i, \boldsymbol{A}_i)$ and labels $\boldsymbol{L}_i$; in particular, the analyst never sees the raw anchor $\boldsymbol{A}$ or the secret bases $\boldsymbol{F}_i$, and thus cannot directly invert user-side transformations.

The shared anchor $\boldsymbol{A} \in \mathbb{R}^{a \times m}$ is *common across users*, but its *origin* (public vs. synthetic) and its *visibility* (whether the analyst can access the exact $\boldsymbol{A}$ used by the users) are distinct design choices. In the default DC threat model, users share $\boldsymbol{A}$ among themselves (or share a PRNG seed that deterministically generates $\boldsymbol{A}$), while the analyst receives only the transformed anchors $\boldsymbol{A}_i := \boldsymbol{A}\boldsymbol{F}_i$ (Algorithm 1, Step 5). When we say that an anchor is "public" we mean *non-sensitive / not derived from any user's private records*; the analyst need not be assumed to possess the exact $\boldsymbol{A}$ instance unless stated otherwise. If the analyst *does* know $\boldsymbol{A}$ (e.g., a truly public anchor dataset), the setting falls outside the semi-honest analyst model and is discussed under the collusion model.

Under the semi-honest model, the privacy guarantees and limitations of this work coincide with those of the underlying DC framework. In particular, as long as all participants remain semi-honest and non-colluding, the subsequent

privacy results do not depend on the number of users c; adding more semi-honest users does not, by itself, reveal additional information about any individual dataset.

The following two theorems are direct restatements, in our notation, of the privacy results established for DC in [8]. They are included for completeness and are not claimed as new contributions.

Theorem 2.1. ***Privacy Against Semi-Honest Users*** (Adapted from Theorem 1 in [8]).
Any semi-honest user i in the DC framework cannot infer the private dataset $\boldsymbol{X}_j$ of any other user $j \neq i$.

Proof. A semi-honest user i has access only to its own private dataset $(\boldsymbol{X}_i, \boldsymbol{L}_i)$, the shared anchor dataset $\boldsymbol{A}$, and the collaboratively trained model h. Information about another user $j \neq i$ is available only through the learned model h and, implicitly, through the transformed representations $\hat{\boldsymbol{X}}_j$ that contributed to training.

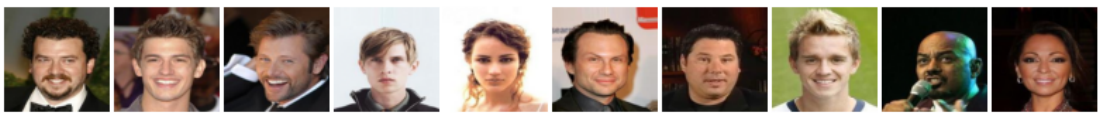
The anchor dataset $\boldsymbol{A}$ is either publicly sourced or synthetically generated and does not contain any information derived from $\boldsymbol{X}_j$. Moreover, the transformed data for user j is given by $\hat{\boldsymbol{X}}_j = \boldsymbol{X}_j\boldsymbol{F}_j\boldsymbol{G}_j$, where both transformation matrices $\boldsymbol{F}_j$ and $\boldsymbol{G}_j$ are unknown to user i. Without access to $\boldsymbol{F}_j$ and $\boldsymbol{G}_j$, user i cannot invert the transformation or recover meaningful information about $\boldsymbol{X}_j$. Hence, a semi-honest user cannot infer the private dataset of any other user. □

Theorem 2.2. ***Privacy Against a Semi-Honest Analyst*** (Adapted from Theorem 2 in [8]).
A semi-honest analyst in the DC framework cannot infer the private dataset $\boldsymbol{X}_j$ of any user j.

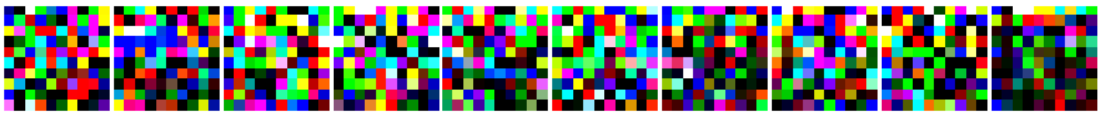
Proof. The semi-honest analyst has access only to the outputs of the user-side linear transformations, namely $\tilde{\boldsymbol{X}}_j = \boldsymbol{X}_j\boldsymbol{F}_j$ and $\boldsymbol{A}_j = \boldsymbol{A}\boldsymbol{F}_j$ for each user j. However, both the transformation matrices $\boldsymbol{F}_j$ and the anchor $\boldsymbol{A}$ are unknown to the analyst. Thus, the analyst observes only pairs $(\tilde{\boldsymbol{X}}_j, \boldsymbol{A}_j)$ with an unknown common factor $\boldsymbol{F}_j$, and lacks sufficient information to recover $\boldsymbol{X}_j$ from $\tilde{\boldsymbol{X}}_j$. Therefore, a semi-honest analyst cannot infer any user's private dataset. □

Contemporary DC techniques typically employ orthonormal bases to transform the private data. Intuitively, enforcing orthonormality on the secret bases preserves geometric properties (e.g., distances and angles) but does not aid reconstruction in the absence of the bases themselves. Consequently, orthonormality does not weaken the privacy guarantees under the semi-honest model.

To visually support this intuition, Fig. 2 presents a simple illustrative example on CelebA [24]. We demonstrate that orthonormal projections significantly degrade visual recognizability, and the degree of obfuscation is comparable to that achieved by general (non-orthogonal) random projections. A more detailed visual privacy analysis appears later in § 7.



(a) Original images



(b) Orthogonally projected images ($\ell = 300$)

(c) Randomly projected images ($\ell = 300$)

Figure 2: Visual privacy verification using CelebA [24]. Original images (panel (a)) compared to images after orthonormal projections (panel (b)) and random non-orthogonal projections (panel (c)). Both transformations strongly obfuscate the visual content, illustrating that the orthonormality assumption does not compromise visual privacy relative to general projections.

Remarks on collusion model. In practice, as the number of users increases, it becomes more likely that some participants deviate from the semi-honest assumption. A natural extension of the DC threat model therefore considers the possibility of *malicious collusion* between participants. In particular, when a user i colludes with the analyst, the colluding parties jointly observe another user j's transformed data, i.e., $\boldsymbol{A}_j$ and $\tilde{\boldsymbol{X}}_j$, as well as the raw anchor $\boldsymbol{A}$. Since $\boldsymbol{A}$ has full column rank, this enables reconstruction of the secret basis via the Moore–Penrose pseudoinverse [15], $\boldsymbol{F}_j = \boldsymbol{A}^{\dagger}\boldsymbol{A}_j$. As argued in [8], because $\boldsymbol{F}_j \in \mathbb{R}^{m\times\ell}$ with $m > \ell$, the matrix $\boldsymbol{F}_j$ acts as a projection, and choosing a smaller latent dimension ℓ makes it substantially harder to reconstruct the original data $\boldsymbol{X}_j$ from $\tilde{\boldsymbol{X}}_j = \boldsymbol{X}_j\boldsymbol{F}_j$, consistent with the guarantees of ε-DR privacy [25].

To mitigate such collusion-based risks and align with stronger privacy standards, differential-privacy-based mechanisms and procedures to eliminate identifiability have been integrated into DC, as demonstrated in [9, 10]. These techniques provide additional protection under threat models that go beyond semi-honest behavior; we refer interested readers to these works for a detailed treatment of stronger adversaries.

We emphasize that this study neither introduces new privacy vulnerabilities nor attempts to strengthen the existing privacy guarantees of DC. Our focus is instead on the *concordance* properties of DC and on improving its computational efficiency, as discussed in § 5. The ODC methodology is fully compatible with enhanced DC variants that offer stronger privacy protections [9, 10], but a comprehensive formal analysis and empirical evaluation of such combinations are left for future work.

2.3. Communication Overhead

In this subsection, we quantify the communication overhead incurred by DC and compare it with standard FL. As illustrated in **Algorithm** 1, DC requires at most three communication phases:

1. A one-off distribution of a common anchor dataset to all users (Step 2).
2. A single uplink from each user to the analyst, containing transformed representations (Step 5).
3. A final downlink from the analyst to each user, comprising the trained model and the change-of-basis matrix (Step 15).

Let q denote the number of bits per scalar element (e.g., q=32 for FP32 and q=16 for FP16), and let N denote the number of parameters in the downstream model h. We write c for the number of users, n_i for the private sample count at user i, $\bar{n} := \frac{1}{c}\sum_{i=1}^{c} n_i$ for the average sample count, a for the anchor size, m for the input dimension, and ℓ for the latent dimension.

For each user $i \in [c]$, the *uplink* payload consists of the transformed representations of both its private dataset and the anchor dataset:

$$B_i^{\mathrm{DC}\uparrow} = \frac{(n_i + a)\,\ell\,q}{8} \quad \text{[bytes]}. \tag{10}$$

The corresponding *downlink* from the analyst to user i contains the trained model h and the (change-of-basis) alignment matrix $\boldsymbol{G}_i \in \mathbb{R}^{\ell\times\ell}$:

$$B_i^{\mathrm{DC}\downarrow} = \frac{(\ell^2 + N)\,q}{8} \quad \text{[bytes]}. \tag{11}$$

Let

$$B_A := \frac{a\,m\,q}{8} \quad \text{[bytes]} \tag{12}$$

denote the size of the raw anchor matrix $\boldsymbol{A} \in \mathbb{R}^{a\times m}$ for one copy. The *aggregate* cross-institution traffic required to make $\boldsymbol{A}$ available can be written as

$$B^{\mathrm{anchor}} = \gamma\, B_A = \gamma\,\frac{a\,m\,q}{8} \quad \text{[bytes]}, \tag{13}$$

where $\gamma \geq 0$ is a topology-dependent *replication factor* (how many cross-silo copies traverse links).

In cross-silo settings, anchor distribution is typically unicast (so $\gamma \approx c$), unless a public/seeded anchor is used ($\gamma = 0$). Importantly, this anchor cost is *one-off*.

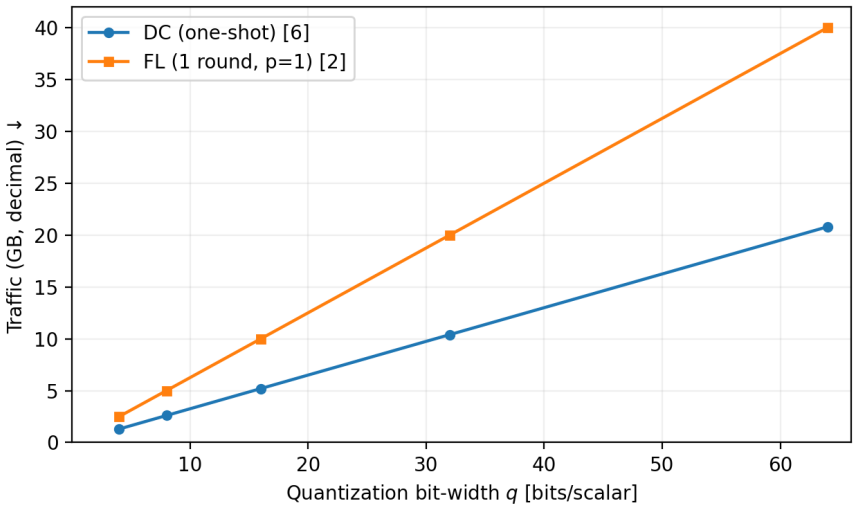


Figure 3: Absolute communication volume versus quantization bit-width q for the healthcare example ($\gamma = c$).

Summing (10), (11), and (13), the aggregate DC traffic becomes

$$\begin{aligned} B^{\mathrm{DC}} &= \sum_{i=1}^{c} \left(B_i^{\mathrm{DC}\uparrow} + B_i^{\mathrm{DC}\downarrow} \right) + B^{\mathrm{anchor}} \\ &= \frac{q}{8} \Big[c((\bar{n} + a)\,\ell + (\ell^2 + N)) + \gamma\, a\, m \Big] \quad \text{[bytes]}. \end{aligned} \tag{14}$$

DC is a *single-round* protocol: the cost (14) is incurred once (plus the one-off anchor distribution if $\gamma > 0$).

Let R denote the number of FL rounds and $p \in (0, 1]$ the per-round participation fraction (constant across rounds). Each selected participant uploads and downloads the full model once per round, thus transmitting $2N$ scalars per round. The cumulative FL traffic is

$$B^{\mathrm{FL}} = 2\, R\, p\, c\, \frac{N\, q}{8} \quad \text{[bytes]}. \tag{15}$$

Setting $B^{\mathrm{DC}} = B^{\mathrm{FL}}$ and solving for R yields

$$R^* = \frac{(\bar{n} + a)\,\ell}{2pN} + \frac{\ell^2 + N}{2pN} + \frac{\gamma\, a\, m}{2p\, c\, N}. \tag{16}$$

Thus, DC is more communication-efficient whenever $R \geq \lceil R^* \rceil$. Note that q cancels exactly in (16); the dependence on c appears only through the anchor term and disappears in the common unicast case $\gamma \approx c$ (e.g., cold start or coordinator distribution).

Equation (16) implies that $R^* \propto 1/p$ and decreases with model size N. While R^* is independent of the bit-width q, the *absolute* traffic scales linearly with q:

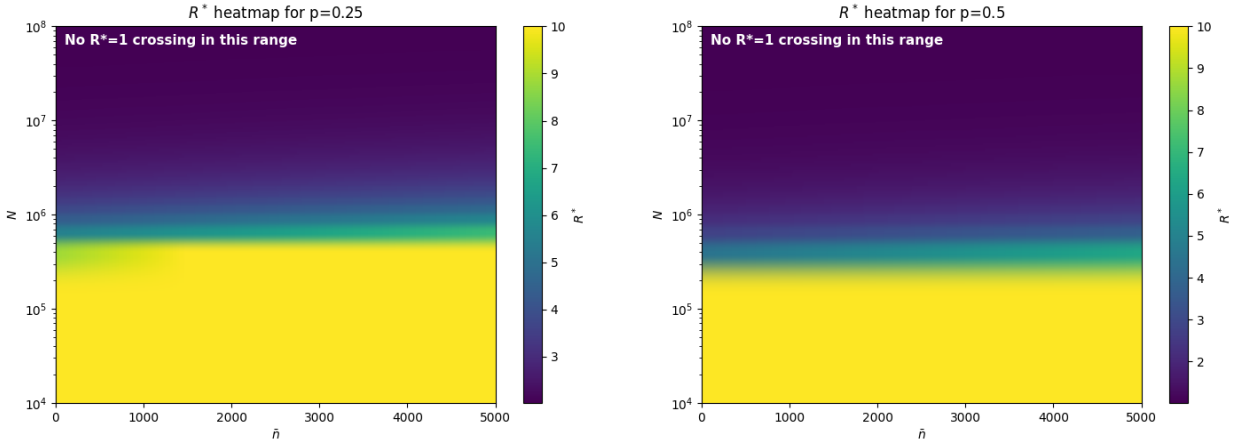
$$B^{\mathrm{DC}}(q) = \frac{q}{q_0} B^{\mathrm{DC}}(q_0), \qquad B^{\mathrm{FL}}(q) = \frac{q}{q_0} B^{\mathrm{FL}}(q_0), \tag{17}$$

where q_0 is just the baseline quantization bit-width.

For the numerical setup below, Fig. 3 plots B^{DC} and B^{FL} as functions of q. The heatmaps in Fig. 4 illustrate how R^* shifts with respect to $\bar{n}$ and N (for several values of p), while Fig. 5 presents one-dimensional slices, showing R^* versus N (for multiple p) and R^* versus p (for multiple N).

Let β denote the effective *bottleneck* goodput (bits/s) and τ the RTT (seconds). A simple engineering estimate for end-to-end transfer time (ignoring compute and assuming payload dominates) is

$$T^{\mathrm{DC}} \approx \frac{8\, B^{\mathrm{DC}}}{\beta} + 3\tau, \qquad T^{\mathrm{FL}} \approx \frac{8\, B^{\mathrm{FL}}}{\beta} + 2R\tau, \tag{18}$$



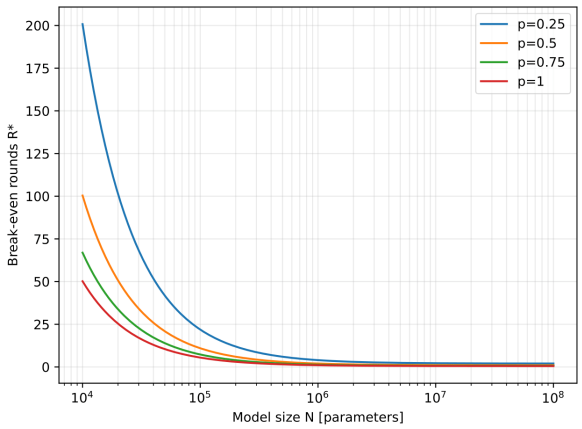
(a) $p = 0.25$

(b) $p = 0.5$

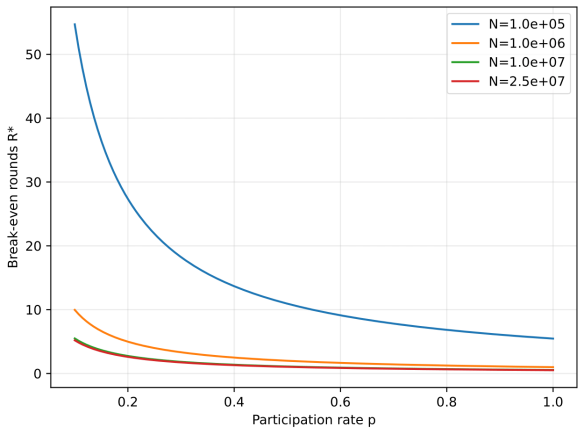
(c) $p = 0.75$

(d) $p = 1.0$

Figure 4: Heatmaps of the threshold number of FL rounds R^* for different participation rates p. Brighter regions correspond to larger R^*, i.e., more FL rounds are required for DC to be more communication-efficient. The white line indicates the break-point for $R^* = 1$, i.e., regions above this line mean a single FL round costs more than DC.



(a) R^* as a function of model size N for $p \in \{0.25, 0.5, 0.75, 1.0\}$ (fixed $\bar{n}$).



(b) R^* as a function of participation p for several N (fixed $\bar{n}$).

Figure 5: Sensitivity curves for the break-even FL rounds R^* (example settings: $a = 10^3$, $m = 784$, $\ell = 100$, $\gamma = c$).

Table 1: Unit conversions. We report GB in decimal units (10^9 bytes) when translating to network time.

Unit	Decimal (SI)	Binary (IEC)
KB / KiB	10^3 B	2^{10} B
MB / MiB	10^6 B	2^{20} B
GB / GiB	10^9 B	2^{30} B

where the additive RTT terms reflect the three DC phases and the two-message-per-round structure of synchronous FL. If clients are the bottleneck rather than the coordinator, one can replace B^{DC} (resp. B^{FL}) by the maximum per-client transmitted bytes in the corresponding phase.

To avoid ambiguity, Table 1 lists the decimal (network) and binary (storage) unit conventions. We report GB in decimal units (10^9 bytes) when translating communication volume to network time.

Consider $c = 100$ hospitals, each with $n_i = 10^3$ samples; anchor size $a = 10^3$; input dimension $m = 784$; latent dimension $\ell = 100$; and a ResNet-50 with $N \simeq 2.5 \times 10^7$ parameters. Assuming 32-bit quantization ($q = 32$) and a cold-start/unicast anchor distribution ($\gamma = c$), (14) gives

$$B^{\mathrm{DC}} \approx 1.04 \times 10^{10} \text{ bytes} \approx 10.4 \text{ GB}, \tag{19}$$

while one round of FL with full participation ($p = 1$) incurs

$$B^{\mathrm{FL}} \approx 2.0 \times 10^{10} \text{ bytes} \approx 20 \text{ GB}. \tag{20}$$

Thus, DC achieves nearly a 50% reduction in communication even compared to a *single* full-participation FL round. (If the anchor is public/seeded, $\gamma = 0$ and B^{DC} decreases slightly by removing the one-off anchor term.)

Under partial participation, say $p = 0.1$, (16) yields $R^* \approx 5.2$ (for $\gamma = c$): FL must run for more than about five rounds before its cumulative traffic exceeds that of DC. Since practical FL deployments often require tens to hundreds of rounds to converge, DC is expected to yield substantial communication savings in realistic cross-silo settings.

3. Related Works

3.1. Procrustes methods in multi-view alignment

The orthogonal Procrustes problem (OPP) [14] is a classical tool for aligning two matrices by an orthogonal transformation. In multi-view alignment, OPP and its multi-set generalizations underpin a large family of methods

that seek a common latent representation shared across views. In functional neuroimaging, *hyperalignment* aligns subject-specific response matrices to a common template by solving a multi-set orthogonal Procrustes problem, under the assumption that subject-specific representational spaces are related by orthogonal transformations [26]. Kernel Hyperalignment extends this framework by solving a regularized multi-set OPP in a reproducing kernel Hilbert space, thereby accommodating nonlinear similarities while retaining the orthogonality constraint on each subject-specific map [27]. Closely related generalized orthogonal Procrustes problems (GOPP) seek a collection of orthogonal matrices that best align multiple point clouds, and recent work has analyzed semidefinite relaxations and generalized power methods with sharp recovery guarantees under signal-plus-noise models [28].

In multi-view representation learning and clustering, Procrustes-based alignment is also widely used. Many methods learn view-specific embeddings $\boldsymbol{X}_i$ together with orthogonal transforms $\boldsymbol{R}_i$ and a consensus representation $\boldsymbol{Z}$ by solving objectives of the form

$$\min_{\boldsymbol{Z},\boldsymbol{R}_i\in\mathcal{O}(\ell)} \sum_i \left\|\boldsymbol{X}_i\boldsymbol{R}_i - \boldsymbol{Z}\right\|_{\mathrm{F}}^2, \tag{21}$$

with variants that introduce adaptive view weights, Grassmannian constraints, or anchor-based parameterizations. For example, Multiview Clustering via Adaptively Weighted Procrustes (AWP) explicitly formulates clustering as a weighted Procrustes averaging problem over spectral embeddings from different views [29], while Multi-view Clustering with Adaptive Procrustes on Grassmann Manifold (MC-APGM) learns an indicator matrix that approximates multiple orthogonal spectral embeddings on the Grassmann manifold, again through an orthogonal Procrustes-type objective [30]. Similar multi-set Procrustes formulations appear in manifold alignment, where independently constructed embeddings of multiple graphs or feature views are brought into correspondence by an orthogonal map [31].

Beyond "classical" multi-view settings, OPP has become a standard primitive for aligning independently trained representation spaces. In natural language processing, Wasserstein–Procrustes objectives are used to align word or sentence embeddings across languages by jointly estimating an optimal transport plan and an orthogonal mapping between embedding spaces [32]. In knowledge-graph representation learning, highly efficient frameworks repeatedly apply closed-form orthogonal Procrustes updates to keep entity and relation embeddings in a shared space while reducing training time and memory footprint by orders of magnitude [33]. In these works, orthogonal transformations are primarily valued for preserving inner products and norms, thereby keeping the geometric structure of each representation space intact while making different embeddings comparable.

Conceptually, our ODC formulation shares the same mathematical backbone as these multi-set Procrustes methods: under our orthonormal-basis assumption on the secret bases in § 5, basis alignment in ODC can be written as

$$\min_{\boldsymbol{Z}\in\mathbb{R}^{a\times\ell},\boldsymbol{G}_i\in\mathcal{O}(\ell)} \quad \sum_{i=1}^{c} \|\boldsymbol{A}_i\boldsymbol{G}_i - \boldsymbol{Z}\|_{\mathrm{F}}^2. \tag{22}$$

which is a multi-view OPP over the anchor representations $\boldsymbol{A}_i$. What fundamentally distinguishes ODC is the *role* it plays and the way we exploit its non-uniqueness. Existing multi-view Procrustes approaches typically fix one view as a template, or implicitly choose a particular Z (e.g., via the SVD of concatenated data) and then interpret the resulting orthogonal transforms as *the* alignment; the fact that the solution set of (21) is closed under right-multiplication by a common orthogonal matrix is treated as a benign identifiability issue and is not analyzed in connection with downstream learning objectives [26, 27, 29, 30, 32, 33].

In contrast, our notion of *orthogonal concordance* 5.2 elevates this indeterminacy to the main object of study. Assuming orthonormal secret bases $\boldsymbol{F}_i$, we show that there exists a common matrix $\boldsymbol{F}$ and orthogonal matrices $\boldsymbol{O}_i$ such that $\boldsymbol{F}_i = \boldsymbol{F}\boldsymbol{O}_i$, and that any Procrustes solution can be written in the form $\boldsymbol{G}_i^\star = \boldsymbol{O}_i^\top\boldsymbol{O}$ for an *arbitrary* global orthogonal matrix $\boldsymbol{O}$. Consequently, all aligned representations $\boldsymbol{F}_i\boldsymbol{G}_i^\star$ collapse to the same product $\boldsymbol{F}\boldsymbol{O}$, and every choice of $\boldsymbol{O}$ in the Procrustes equivalence class yields *identical* downstream predictions for distance-based models (and, empirically, near-identical performance more broadly). This invariance property — that the entire multi-set Procrustes solution set induces the same collaborative model — is what we call orthogonal concordance, and it is specific to the DC setting with orthonormal secret bases and anchor-only access to the data.

Finally, most Procrustes-based multi-view methods assume a centralized optimizer that directly observes all raw views $\boldsymbol{X}_i$ or their feature embeddings and do not model privacy or communication constraints [29, 30, 32, 33]. ODC instead operates in a privacy-preserving DC pipeline, where the analyst only sees intermediate anchor representations $\boldsymbol{A}_i$ and user-transformed data. The goal is not merely to reduce alignment error, but to guarantee that the entire

equivalence class of Procrustes solutions yields the same collaborative model. This theoretical shift—from *finding a good orthogonal alignment* to *characterizing all alignments that are provably equivalent for learning*—is the key difference between classical Procrustes-based multi-view alignment and our concept of orthogonal concordance.

3.2. Privacy-Preserving Machine Learning

Privacy-preserving machine learning (PPML) broadly studies how to learn from sensitive data without directly centralizing raw records. Among the most widely deployed paradigms is *federated learning* (FL), where clients repeatedly run local training and a server aggregates model updates over many communication rounds (e.g., FedAvg) [2]. Because FL communicates gradients/weights, it does not, by itself, provide a formal privacy guarantee; consequently, FL is often paired with *differential privacy* (DP) mechanisms that add calibrated noise to client updates to obtain an (ε, δ)-DP guarantee [3, 4]. DP provides worst-case, distribution-free protection—output distributions remain stable to the inclusion/exclusion of any single record—but can impose a non-trivial privacy–utility tradeoff, especially at strong privacy levels [3].

Data collaboration (DC) offers a complementary PPML design point: instead of iteratively sharing model updates, each party shares *one-shot intermediate representations* obtained by applying a private (secret) transform to its data (and to a shared anchor), after which an analyst aligns representations into a common space for downstream learning [2, 10]. This yields a different theoretical separation from FL/DP:

- **Optimization object:** FL performs distributed optimization of model parameters via iterative aggregation [2], whereas DC centralizes *aligned representations* and can then train arbitrary downstream models in that shared space [6].
- **Communication pattern:** FL typically requires many bidirectional rounds until convergence; DC uses a single uplink of representations (plus a one-time anchor broadcast), which can be advantageous in cross-silo settings with large models [2, 6].
- **Privacy semantics:** DP provides a formal indistinguishability guarantee via noise [3], while DC primarily relies on secrecy of the local transform (often analyzed under semi-honest assumptions) rather than an (ε, δ) guarantee [8].

Our proposed ODC remains within the DC family, retaining DC's one-shot protocol, while constraining alignment to orthogonal transformations, thereby preserving geometric structure and simplifying the analytical alignment process. Finally, DP and DC are not mutually exclusive: DP can be layered on top of released representations to strengthen protection under collusion, and DP-enhanced DC has been explicitly explored [10].

4. Existing Basis Alignment

Focusing on Step 10 of **Algorithm** 1, we now address the central question of DC:

$$\textit{How can we construct the change-of-basis matrices } \boldsymbol{G}_i \textit{ without access to the secret bases } \boldsymbol{F}_i\textit{?} \tag{23}$$

In this section, we review the two prevailing basis-alignment strategies employed in existing DC frameworks—Imakura-DC and Kawakami-DC—and analyze their concordance properties and computational efficiency, measured by FLOP count and peak memory usage.

4.1. Imakura's Basis Alignment (Imakura-DC)

Here, we review the basis-alignment method proposed by Imakura *et al.* [8]. To begin with, we introduce **Assumption** 4.1.

Assumption 4.1. *We impose the following assumptions on the anchor dataset* $\boldsymbol{A} \in \mathbb{R}^{a\times m}$ *and the secret bases* $\boldsymbol{F}_i \in \mathbb{R}^{m\times \ell}$ $(m \geq \ell)$*, for all* $i \in [c]$*:*

1) $\mathrm{rank}(\boldsymbol{A}) = m$*;*

Algorithm 2: Imakura-DC's Basis Alignment Procedure [8]

Input : $\boldsymbol{A}_i \in \mathbb{R}^{a\times\ell}$ for each user $i \in [c]$
Output: $\boldsymbol{G}_i^* \in \mathcal{GL}(\ell)$ for each user $i \in [c]$

begin
  Form the concatenated anchor $[\, \boldsymbol{A}_1 \ \cdots \ \boldsymbol{A}_c \,]$ and compute its top ℓ left singular vectors $\boldsymbol{U}$
  Set the target $\boldsymbol{Z} = \boldsymbol{U}\boldsymbol{R}$ with any $\boldsymbol{R} \in \mathcal{GL}(\ell)$ (often $\boldsymbol{R} = \boldsymbol{I}$)
  Set $\boldsymbol{G}_i^* = \boldsymbol{A}_i^\dagger \boldsymbol{Z}$ for all i
end

2) There exists $\boldsymbol{E}_i \in \mathcal{GL}(\ell) \coloneqq \{\boldsymbol{R} \in \mathbb{R}^{\ell\times\ell} : \mathrm{rank}(\boldsymbol{R}) = \ell\}$ *such that* $\boldsymbol{F}_i = \boldsymbol{F}_1\boldsymbol{E}_i$.

Condition 1) is a common requirement in the DC literature. For instance, selecting $\boldsymbol{A}$ to be a uniformly random matrix (with appropriate dimensions) ensures $\mathrm{rank}(\boldsymbol{A}) = m$, which is standard practice. Condition 2) requires all intermediate representations to span an identical ℓ-dimensional subspace.

The central task for the analyst is to construct suitable change-of-basis matrices $\boldsymbol{G}_i$ that *align* the secret bases $\boldsymbol{F}_i$, even though these bases are never directly observed. Formally, the objective is to identify matrices $\boldsymbol{G}_i \in \mathcal{GL}(\ell)$ for each $i \in [c]$ such that

$$\boldsymbol{F}_1\boldsymbol{G}_1 \ = \ \boldsymbol{F}_2\boldsymbol{G}_2 \ = \ \ldots \ = \ \boldsymbol{F}_c\boldsymbol{G}_c. \tag{24}$$

Note that such $\boldsymbol{G}_i$ exist if and only if condition 2) of **Assumption** 4.1 is met.

Since the analyst has access to the intermediate representations of a common anchor dataset, defined as $\boldsymbol{A}_i = \boldsymbol{A}\boldsymbol{F}_i \in \mathbb{R}^{a\times\ell}$, where $\boldsymbol{A}$ remains consistent across all users, it follows that any set of matrices $\boldsymbol{G}_i$ satisfying (24) necessarily satisfy the following condition:

$$\boldsymbol{A}_1\boldsymbol{G}_1 = \boldsymbol{A}_2\boldsymbol{G}_2 = \cdots = \boldsymbol{A}_c\boldsymbol{G}_c. \tag{25}$$

Consequently, the matrices $\boldsymbol{G}_i$ that satisfy equation (25) are necessarily those that minimize the following optimization problem:

$$\min_{\boldsymbol{G}_i,\boldsymbol{G}_j\in\mathcal{GL}(\ell)} \quad \sum_{i=1}^{c}\sum_{j=1}^{c}\left\|\boldsymbol{A}_i\boldsymbol{G}_i - \boldsymbol{A}_j\boldsymbol{G}_j\right\|_{\mathrm{F}}^2. \tag{26}$$

A crucial observation is that $\mathcal{GL}(\ell)$ is inherently non-compact. This non-compactness poses significant difficulties, specifically that one can construct a sequence of invertible matrices whose singular values decrease, causing the sequence to converge to the zero matrix. Such convergence undermines the stability of the optimization and complicates the identification of meaningful solutions. This ill-posedness can, however, be alleviated by a judicious, *a priori* selection of some target matrix $\boldsymbol{Z} \in \mathbb{R}^{a\times\ell}$. Specifically, define $\boldsymbol{Z} = \boldsymbol{U}\boldsymbol{R}$, where $\boldsymbol{U}$ comprises the top ℓ left singular vectors of the concatenated matrix $\begin{bmatrix}\boldsymbol{A}_1 & \cdots & \boldsymbol{A}_c\end{bmatrix}$, and set an arbitrary invertible matrix $\boldsymbol{R} \in \mathcal{GL}(\ell)$. Problem (26) reduces to:

$$\min_{\boldsymbol{G}_i\in\mathcal{GL}(\ell)} \|\boldsymbol{A}_i\boldsymbol{G}_i - \boldsymbol{Z}\|_{\mathrm{F}}^2. \tag{27}$$

The optimization problem in (27) admits closed-form analytical solutions given by $\boldsymbol{G}_i^* = \boldsymbol{A}_i^\dagger\boldsymbol{Z}$ for each $i \in [c]$. Imakura-DC's basis alignment procedure is summarized in **Algorithm** 2.

4.1.1. Weak Concordance of Imakura-DC

To theoretically assess the downstream performance of the resulting change-of-basis matrices $\boldsymbol{G}_i^*$, we analyze their sufficiency with respect to Eq. (24). We formalize *weak concordance* as follows:

Definition 4.2. *(Weak Concordance)*
The change-of-basis matrices $\boldsymbol{G}_i \in \mathcal{GL}(\ell)$*, for* $i \in [c]$*, satisfy weak concordance if*

$$\boldsymbol{F}_1\boldsymbol{G}_1 = \boldsymbol{F}_2\boldsymbol{G}_2 = \cdots = \boldsymbol{F}_c\boldsymbol{G}_c. \tag{28}$$

Theorem 4.3. *(Weak Concordance of Imakura-DC (cf. [8]))*
Suppose that we observe matrices $\boldsymbol{A}_i = \boldsymbol{A}\boldsymbol{F}_i$, $i \in [c]$, *with* $\boldsymbol{A} \in \mathbb{R}^{a\times m}$ *and* $\boldsymbol{F}_i \in \mathbb{R}^{m\times \ell}$. *Under* ***Assumption*** *4.1, let* $\boldsymbol{U}$ *denote the matrix formed by the top* ℓ *left singular vectors of the concatenation* $[\, \boldsymbol{A}_1 \ \cdots \ \boldsymbol{A}_c \,]$. *Then, for each* $i \in [c]$, *the solution* $\boldsymbol{G}_i^* = \boldsymbol{A}_i^\dagger \boldsymbol{Z}$ *to the optimization problem:*

$$\min_{\boldsymbol{G}_i \in \mathbb{R}^{\ell\times\ell}} \|\boldsymbol{A}_i\boldsymbol{G}_i - \boldsymbol{Z}\|_{\mathrm{F}}^2, \tag{29}$$

where $\boldsymbol{Z} = \boldsymbol{U}\boldsymbol{R}$ *for an arbitrary invertible matrix* $\boldsymbol{R} \in \mathcal{GL}(\ell)$, *is weakly concordant (****Definition*** *4.2).*

Proof. Since all $\boldsymbol{A}_i$ share the same ℓ-dimensional column space, the top ℓ left singular vectors $\boldsymbol{U}$ of $[\, \boldsymbol{A}_1 \ \cdots \ \boldsymbol{A}_c \,]$ also lie in the same column space. Therefore, there exists some invertible matrix $\boldsymbol{Q} \in \mathcal{GL}(\ell)$ such that:

$$\boldsymbol{Z} = \boldsymbol{A}_1\boldsymbol{Q}. \tag{30}$$

Then we can write:

$$\boldsymbol{Z} = \boldsymbol{A}_1\boldsymbol{Q} = \boldsymbol{A}\boldsymbol{F}_1\boldsymbol{E}_i(\boldsymbol{E}_i^{-1}\boldsymbol{Q}) = \boldsymbol{A}_i(\boldsymbol{E}_i^{-1}\boldsymbol{Q}). \tag{31}$$

Hence,

$$\boldsymbol{G}_i^* = \boldsymbol{A}_i^\dagger \boldsymbol{Z} = \boldsymbol{A}_i^\dagger \boldsymbol{A}_i(\boldsymbol{E}_i^{-1}\boldsymbol{Q}) = \boldsymbol{E}_i^{-1}\boldsymbol{Q}, \tag{32}$$

and therefore, we have

$$\boldsymbol{F}_1\boldsymbol{G}_1^* = \cdots = \boldsymbol{F}_c\boldsymbol{G}_c^*, \tag{33}$$

which completes the proof.

□

Theorem 4.3 establishes that the invertible right factor $\boldsymbol{R} \in \mathbb{R}^{\ell\times\ell}$ of the target matrix $\boldsymbol{Z} = \boldsymbol{U}\boldsymbol{R}$ can be chosen arbitrarily while preserving weak concordance. This flexibility naturally prompts the question of whether such arbitrary choices could negatively impact downstream model performance. Unfortunately, the answer is affirmative. Empirical evidence examining this issue is presented in §8. Intuitively, selecting a target matrix that disproportionately emphasizes certain directions within the feature space may adversely affect model accuracy and utility. Although recent studies empirically indicate improved performance when choosing $\boldsymbol{R} = \boldsymbol{I}$ [6], this particular choice remains heuristic and lacks rigorous theoretical justification, suggesting potential suboptimality.

Consequently, the practical utility of **Definition** 4.2 and **Theorem** 4.3 is inherently limited. Indeed, Imakura's basis alignment method exhibits a notable discrepancy between its theoretical guarantees and empirical performance.

4.1.2. FLOPs of Imakura-DC

We adopt standard BLAS/LAPACK floating-point operation (FLOP) counts [34, 35, 36, 37], summarized in Table 2.

Table 2: Standard matrix operations and their approximate floating-point operation (flop) counts.

Operation	Matrix	Approximate FLOPs
Matrix product ($\boldsymbol{C} = \boldsymbol{A}\boldsymbol{B}$)	$\boldsymbol{A} \in \mathbb{R}^{m\times k}$, $\boldsymbol{B} \in \mathbb{R}^{k\times n}$	$\approx 2mkn$
QR factorization (Householder, $n = \min\{m, n\}$)	$\boldsymbol{A} \in \mathbb{R}^{m\times n}$	$\approx 2mn^2 - \frac{2}{3}n^3$
SVD ($n = \min\{m, n\}$)	$\boldsymbol{A} \in \mathbb{R}^{m\times n}$	$\approx 4mn^2 + 8n^3$
Triangular inverse	$\boldsymbol{R} \in \mathbb{R}^{n\times n}$	$\approx \frac{1}{3}n^3$

In **Algorithm** 2, the concatenated anchor matrix

$$[\, \boldsymbol{A}_1 \ \cdots \ \boldsymbol{A}_c \,] \in \mathbb{R}^{a\times c\ell} \tag{34}$$

has size $a \times c\ell$. Let

$$p = \min\{a, c\ell\}, \qquad q = \max\{a, c\ell\}. \tag{35}$$

Using the FLOP counts in Table 2, we estimate the cost of Imakura-DC as follows:

- Computing the SVD of the $a \times c\ell$ matrix $[\, \boldsymbol{A}_1 \ \cdots \ \boldsymbol{A}_c \,]$ costs

$$4qp^2 + 8p^3 \text{ FLOPs.} \tag{36}$$

- The matrix $\boldsymbol{U} \in \mathbb{R}^{a\times\ell}$ contains the top ℓ left singular vectors. Forming $\boldsymbol{Z} = \boldsymbol{UR}$ multiplies an $a \times \ell$ matrix by an $\ell \times \ell$ matrix and costs

$$2a\ell^2 \text{ FLOPs.} \tag{37}$$

- For each user $i \in [c]$, forming the Moore–Penrose pseudoinverse of $\boldsymbol{A}_i \in \mathbb{R}^{a\times\ell}$ via SVD and multiplying it by $\boldsymbol{Z}$ costs

$$4a\ell^2 + 8\ell^3 \ + \ 2a\ell^2 \ = \ 6a\ell^2 + 8\ell^3 \text{ FLOPs.} \tag{38}$$

Thus, the total approximate FLOP count for Imakura-DC is

$$4qp^2 + 8p^3 + 2a\ell^2 + c(6a\ell^2 + 8\ell^3) = 4qp^2 + 8p^3 + (6c + 2)\, a\ell^2 + 8c\ell^3 \text{ FLOPs.} \tag{39}$$

Under the assumption $a > \ell$, this can be summarized in big-O notation as

$$O(\min\{a(c\ell)^2,\ a^2c\ell\}). \tag{40}$$

4.1.3. Peak memory of Imakura-DC

We measure memory in units of real scalars. The c matrices $\boldsymbol{A}_i \in \mathbb{R}^{a\times\ell}$ can be viewed as a single concatenated anchor

$$\bar{\boldsymbol{A}} = [\, \boldsymbol{A}_1 \ \cdots \ \boldsymbol{A}_c \,] \in \mathbb{R}^{a\times c\ell}, \tag{41}$$

which already requires $ac\ell$ scalars to store. Let

$$p = \min\{a, c\ell\}. \tag{42}$$

We now bound the peak memory footprint of **Algorithm** 2 by inspecting each step. We assume $a > \ell$ throughout.

- **SVD of the concatenated anchor.** During the computation of the SVD of $\bar{\boldsymbol{A}}$ we store $\bar{\boldsymbol{A}}$ itself ($ac\ell$ scalars), the top ℓ left singular vectors $\boldsymbol{U} \in \mathbb{R}^{a\times\ell}$ ($a\ell$ scalars), and an SVD workspace of size $O(p^2)$ scalars. Since $p = \min\{a, c\ell\}$, we have $p^2 \le ac\ell$, and clearly $a\ell \le ac\ell$, so this step uses

$$ac\ell + O(a\ell + p^2) = O(ac\ell). \tag{43}$$

- **Forming the target $\boldsymbol{Z} = \boldsymbol{UR}$.** Next, we form the target $\boldsymbol{Z} = \boldsymbol{UR}$ with $\boldsymbol{Z} \in \mathbb{R}^{a\times\ell}$ and $\boldsymbol{R} \in \mathcal{GL}(\ell)$, which require $a\ell$ and ℓ^2 scalars, respectively. At this point, we still store $\bar{\boldsymbol{A}}$ and $\boldsymbol{U}$, so the memory usage is

$$ac\ell + 2a\ell + \ell^2 = O(ac\ell). \tag{44}$$

  (The SVD workspace from the previous step can be released before forming $\boldsymbol{Z}$.)

- **Per-user pseudoinverse and alignment.** For each $i \in [c]$, we form $\boldsymbol{A}_i^\dagger$ via the SVD of $\boldsymbol{A}_i \in \mathbb{R}^{a\times\ell}$ and multiply by $\boldsymbol{Z}$. This per-user computation requires $O(a\ell + \ell^2)$ scalars of workspace, in addition to storing $\bar{\boldsymbol{A}}$ and $\boldsymbol{Z}$. Storing all outputs $\boldsymbol{G}_i^*$, where $\boldsymbol{G}_i^* \in \mathbb{R}^{\ell\times\ell}$, costs $c\ell^2$ scalars. Thus, near the end of the loop, we have

$$ac\ell + O(a\ell + c\ell^2). \tag{45}$$

Since $ac\ell \ge a\ell$ for all $c \ge 1$ and, under the natural regime $a \ge \ell$, also $ac\ell \ge c\ell^2$, the peak memory requirement is dominated by storing the concatenated anchor:

$$ac\ell + O(a\ell + c\ell^2) = \Theta(ac\ell). \tag{46}$$

Algorithm 3: Kawakami-DC's Basis Alignment Procedure [12]

Input : $\boldsymbol{A}_i \in \mathbb{R}^{a\times\ell}$ for each user $i \in [c]$
Output: $\boldsymbol{G}_i^* \in \mathbb{R}^{\ell\times\ell}$ for each user $i \in [c]$

begin
  for $i \in [c]$ **do**
    Compute a thin QR factorization $\boldsymbol{A}_i = \boldsymbol{Q}_i\boldsymbol{R}_i$
  end
  Form the concatenated matrix $\boldsymbol{W}_Q \coloneqq [\,\boldsymbol{Q}_1\ \cdots\ \boldsymbol{Q}_c\,] \in \mathbb{R}^{a\times c\ell}$
  Compute the SVD $\boldsymbol{W}_Q = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top$
  Let $\boldsymbol{V}_\ell$ contain the ℓ right singular vectors of $\boldsymbol{W}_Q$ associated with the largest singular values
  for $k \in \{1, 2, \ldots, \ell\}$ **do**
    Set v'_k to the k-th column of $\boldsymbol{V}_\ell$ and partition v'_k into blocks $\hat{g}_{i,k} \in \mathbb{R}^\ell$ for all $i \in [c]$
  end
  for $i \in [c]$ **do**
    Set $\boldsymbol{G}_i^* \coloneqq [\,\boldsymbol{R}_i^{-1}\hat{g}_{i,1}\ \cdots\ \boldsymbol{R}_i^{-1}\hat{g}_{i,\ell}\,]$
  end
end

4.2. Kawakami's Basis Alignment (Kawakami-DC)

Imakura–DC formulates basis alignment by introducing an explicit target matrix that each user must match. Kawakami *et al.* [12] instead propose a target-free formulation that directly aligns the user-specific anchor representations. In the setting of Problem (26), the main difficulty is that the feasible set is non-compact: without additional constraints, one can rescale the change-of-basis matrices so that their singular values decay and the objective is minimized by sequences converging to the zero matrix.

To avoid this degeneracy, Kawakami *et al.* [12] introduce column-wise normalization constraints. Kawakami–DC seeks matrices $\boldsymbol{G}_i$ that solve

$$\begin{aligned} \min_{\boldsymbol{G}_i\in\mathbb{R}^{\ell\times\ell}} \quad & \sum_{i=1}^{c}\sum_{j=1}^{c}\left\|\boldsymbol{A}_i\boldsymbol{G}_i - \boldsymbol{A}_j\boldsymbol{G}_j\right\|_{\mathrm{F}}^2 \\ \text{s.t.} \quad & \sum_{i=1}^{c}\left\|\boldsymbol{A}_i g_{i,k}\right\|_2^2 = 1, \qquad k \in \{1, 2, \ldots, \ell\}, \end{aligned} \tag{47}$$

where $g_{i,k}$ denotes the k-th column of $\boldsymbol{G}_i$.

The constraint fixes the global scale of the k-th aligned feature across all users, as measured on the anchor, while leaving its direction free. Under the standard DC assumption that each $\boldsymbol{A}_i$ has full column rank, the constraint does not restrict which directions $g_{i,k}$ are admissible: for any nonzero collection of $g_{i,k}$, we can always rescale them to satisfy:

$$\sum_{i=1}^{c}\left\|\boldsymbol{A}_i g_{i,k}\right\|_2^2 = 1. \tag{48}$$

For Problem (47), Kawakami *et al.* [12] show that the matrices $\boldsymbol{G}_i$ can be computed efficiently via a QR-SVD-based algorithm (**Algorithm** 3).

Importantly, the primary role of this formulation is to exclude the trivial all-zero solution; it does *not* guarantee that the resulting matrices $\boldsymbol{G}_i$ are invertible (and thus they need not constitute genuine change-of-basis matrices). Moreover, the concordance properties of Kawakami-DC remain unknown and constitute an open problem for future work.

4.2.1. FLOPs of Kawakami-DC

In Kawakami-DC, each anchor matrix $\boldsymbol{A}_i \in \mathbb{R}^{a\times\ell}$ is factorized as $\boldsymbol{A}_i = \boldsymbol{Q}_i\boldsymbol{R}_i$, and the orthonormal factors are concatenated to form

$$\boldsymbol{W}_Q = [\,\boldsymbol{Q}_1\ \cdots\ \boldsymbol{Q}_c\,] \in \mathbb{R}^{a\times c\ell}. \tag{49}$$

Let

$$p = \min\{a, c\ell\}, \qquad q = \max\{a, c\ell\}. \tag{50}$$

Using the FLOP counts in Table 2, we estimate the cost of Kawakami-DC (**Algorithm** 3) as follows:

- For each $\boldsymbol{A}_i \in \mathbb{R}^{a\times\ell}$, the thin QR factorization $\boldsymbol{A}_i = \boldsymbol{Q}_i\boldsymbol{R}_i$ costs

$$2a\ell^2 - \frac{2}{3}\ell^3 \text{ FLOPs.} \tag{51}$$

  Summed over all c users, the total becomes

$$c\Big(2a\ell^2 - \frac{2}{3}\ell^3\Big). \tag{52}$$

- Computing the SVD of the concatenated matrix $\boldsymbol{W}_Q \in \mathbb{R}^{a\times c\ell}$ costs

$$4qp^2 + 8p^3 \text{ FLOPs.} \tag{53}$$

- For each user $i \in [c]$, we explicitly compute the inverse $\boldsymbol{R}_i^{-1}$ of the upper triangular matrix $\boldsymbol{R}_i$. This costs

$$\frac{1}{3}\ell^3 \text{ FLOPs.} \tag{54}$$

  Then, for each $k \in [\ell]$, we recover $g_{i,k}$ via the matrix–vector multiplication $\boldsymbol{R}_i^{-1}\hat{g}_{i,k}$, which costs ℓ^2 FLOPs per column. Since there are ℓ columns, the total per user is

$$\frac{1}{3}\ell^3 + \ell^3 = \frac{4}{3}\ell^3. \tag{55}$$

  Summed over all users, the recovery step costs

$$\frac{4}{3}c\,\ell^3. \tag{56}$$

Thus, the total approximate FLOP count for Kawakami-DC is

$$4qp^2 + 8p^3 + c\Big(2a\ell^2 - \frac{2}{3}\ell^3\Big) + \frac{4}{3}c\,\ell^3 = 4qp^2 + 8p^3 + c\left(2a\ell^2 + \frac{2}{3}\ell^3\right) \text{ FLOPs.} \tag{57}$$

Under the assumption $a > \ell$, this can be summarized in big-O notation as

$$O(\min\{a(c\ell)^2,\, a^2c\ell\}). \tag{58}$$

4.2.2. Peak memory of Kawakami-DC

We measure memory in units of real scalars. The c matrices $\boldsymbol{A}_i \in \mathbb{R}^{a\times\ell}$ are again stored explicitly, requiring

$$ac\ell \tag{59}$$

scalars in total. Let

$$p = \min\{a,\, c\ell\}. \tag{60}$$

We now inspect each step of the Kawakami-DC procedure and bound its peak memory footprint.

- **QR factorizations of $\boldsymbol{A}_i$.** Each $\boldsymbol{A}_i$ is factorized as $\boldsymbol{A}_i = \boldsymbol{Q}_i\boldsymbol{R}_i$, where $\boldsymbol{Q}_i \in \mathbb{R}^{a\times\ell}$ and $\boldsymbol{R}_i \in \mathbb{R}^{\ell\times\ell}$. While computing the QR factorization, we store $\boldsymbol{A}_i$ itself ($a\ell$ scalars), along with the Householder vectors and a workspace of size $O(a\ell)$ scalars. Since $ac\ell$ dominates $a\ell$, this step fits within

$$ac\ell + O(a\ell) = O(ac\ell). \tag{61}$$

Table 3: Theoretical comparison between contemporary DC methods and the proposed ODC framework.

	Imakura-DC	**Kawakami-DC**	**ODC** (this work)
Objective	$\min_{\boldsymbol{G}_i \in \mathbb{R}^{\ell\times\ell}} \|\boldsymbol{A}_i\boldsymbol{G}_i - \boldsymbol{Z}\|_{\mathrm{F}}^2$	$\min_{\boldsymbol{G}_i \in \mathbb{R}^{\ell\times\ell}} \sum_{i,j\in[c]} \|\boldsymbol{A}_i\boldsymbol{G}_i - \boldsymbol{A}_j\boldsymbol{G}_j\|_{\mathrm{F}}^2$	$\min_{\boldsymbol{G}_i \in \mathbb{R}^{\ell\times\ell}} \sum_{i,j\in[c]} \|\boldsymbol{A}_i\boldsymbol{G}_i - \boldsymbol{A}_j\boldsymbol{G}_j\|_{\mathrm{F}}^2$
Constraint	$\boldsymbol{G}_i \in \mathcal{GL}(\ell)$	$\sum_{i=1}^{c} \|\boldsymbol{A}_i g_{i,k}\|_2^2 = 1$	$\boldsymbol{G}_i \in O(\ell)$
Solution Property	Ambiguous up to a common right invertible transform	Not necessarily invertible	Ambiguous up to a common right orthogonal transform
Concordance	Weak concordance	Unknown	Orthogonal concordance
FLOPs	$O(\min\{a(c\ell)^2,\ a^2c\ell\})$	$O(\min\{a(c\ell)^2,\ a^2c\ell\})$	$O(ac\ell^2)$

- **Storing all $\boldsymbol{Q}_i$ and $\boldsymbol{R}_i$.** After QR has been completed for all users, we overwrite each $\boldsymbol{A}_i$ with $\boldsymbol{Q}_i$ and store

$$\boldsymbol{Q}_1, \ldots, \boldsymbol{Q}_c \in \mathbb{R}^{a\times\ell}, \qquad \boldsymbol{R}_1, \ldots, \boldsymbol{R}_c \in \mathbb{R}^{\ell\times\ell}. \tag{62}$$

  This costs

$$c(a\ell) + c\ell^2 = ac\ell + c\ell^2. \tag{63}$$

  Since $a > \ell$, we have $ac\ell > c\ell^2$, hence this stage also requires $O(ac\ell)$ scalars.

- **SVD of the concatenated matrix $\boldsymbol{W}_Q = [\,\boldsymbol{Q}_1\ \cdots\ \boldsymbol{Q}_c\,]$.** The concatenated matrix $\boldsymbol{W}_Q \in \mathbb{R}^{a\times c\ell}$ requires $ac\ell$ scalars to store. The SVD workspace uses $O(p^2)$ scalars with $p = \min\{a, c\ell\} \le ac\ell$. Thus, during SVD, we store

$$ac\ell + O(p^2) = O(ac\ell). \tag{64}$$

- **Recovery of the vectors $g_{i,k}$.** From the SVD, we obtain the blocks $\hat{g}_{i,k} \in \mathbb{R}^{\ell}$. To recover the original vectors $g_{i,k}$ we solve $\boldsymbol{R}_i g_{i,k} = \hat{g}_{i,k}$ by back substitution. Each solve requires $O(\ell^2)$ scalars of temporary workspace. Since we already store $\boldsymbol{W}_Q$ (or the $\boldsymbol{Q}_i$) and $\boldsymbol{R}_i$, the memory during this step is

$$ac\ell + O(\ell^2) = O(ac\ell). \tag{65}$$

  Storing all $\boldsymbol{G}_i^* \in \mathbb{R}^{\ell\times\ell}$ costs $c\ell^2$ scalars, which is dominated by $ac\ell$.

Combining all contributions, the peak memory usage is dominated by storing the concatenated $\boldsymbol{Q}$ matrices (equivalently, the anchor data), giving:

$$ac\ell + O(a\ell + c\ell^2) = \Theta(ac\ell). \tag{66}$$

The term $ac\ell$ dominates all others, so the peak memory of Kawakami-DC (QR–SVD) matches that of Imakura-DC.

5. Proposed Basis Alignment

Here, we present our proposed basis-alignment procedure, *Orthonormal Data Collaboration (ODC)*. Our theoretical findings, along with a comparison to existing DC alignment methods, are summarized in Table 3. In this section, we proceed as follows:

1. We show that, under **Assumption** 5.1, the basis-alignment problem naturally reduces to the classical Orthogonal Procrustes Problem, which admits a closed-form solution.
2. We address the instability inherent in the notion of weak concordance and introduce a refined notion, *orthogonal concordance*.
3. We prove that the ODC basis-alignment procedure satisfies orthogonal concordance.
4. We analyze the computational efficiency of ODC and show that it reduces the alignment time complexity from $O\big(\min\{a(c\ell)^2,\ a^2c\ell\}\big)$ for contemporary DC methods to $O(ac\ell^2)$.

We begin by introducing **Assumption** 5.1.

Assumption 5.1. *We impose the following conditions on the anchor dataset* $\boldsymbol{A} \in \mathbb{R}^{a\times m}$ *and the secret bases* $\boldsymbol{F}_i \in \mathbb{R}^{m\times \ell}$ $(m \geq \ell)$*, for all* $i \in [c]$*:*

1) $\operatorname{rank}(\boldsymbol{A}) = m$*;*

2) There exists $\boldsymbol{E}_i \in \mathcal{GL}(\ell)$ *such that* $\boldsymbol{F}_i = \boldsymbol{F}_1\boldsymbol{E}_i$*;*

3) $\boldsymbol{F}_i^\top \boldsymbol{F}_i = \boldsymbol{I}$*.*

Conditions 1) and 2) coincide with those in **Assumption** 4.1. We additionally impose condition 3), which requires the secret bases to be orthonormal within the ODC framework. It is noteworthy that standard basis-selection methods such as PCA and SVD naturally yield orthonormal bases. Furthermore, to the best of our knowledge, all existing DC applications can readily accommodate this additional orthonormality constraint.

5.1. Basis Alignment as the Orthogonal Procrustes Problem

Similarly to Imakura-DC, our goal is to identify $\boldsymbol{G}_i \in \mathcal{GL}(\ell)$ for each $i \in [c]$ that satisfy Eq. (24) without knowledge of the secret bases $\boldsymbol{F}_i$.

Now, from conditions 2) and 3), it immediately follows that $\boldsymbol{E}_i \in \mathcal{GL}(\ell)$ are orthogonal (i.e., $\boldsymbol{E}_i \in \mathcal{O}(\ell)$), because:

$$\boldsymbol{F}_i^\top \boldsymbol{F}_i = (\boldsymbol{F}_1\boldsymbol{E}_i)^\top(\boldsymbol{F}_1\boldsymbol{E}_i) = \boldsymbol{E}_i^\top \boldsymbol{F}_1^\top \boldsymbol{F}_1 \boldsymbol{E}_i = \boldsymbol{E}_i^\top \boldsymbol{E}_i = \boldsymbol{I}_\ell. \tag{67}$$

Therefore, we can equivalently write our goal as to identify $\boldsymbol{G}_i \in \mathcal{O}(\ell)$ for each $i \in [c]$ that satisfy Eq. (24) without knowledge of the secret bases $\boldsymbol{F}_i$.

Given the analyst has access to $\boldsymbol{A}_i = \boldsymbol{A}\boldsymbol{F}_i \in \mathbb{R}^{a\times\ell}$, $\boldsymbol{G}_i \in \mathcal{O}(\ell)$ necessarily satisfy:

$$\boldsymbol{A}_1\boldsymbol{G}_1 = \boldsymbol{A}_2\boldsymbol{G}_2 = \cdots = \boldsymbol{A}_c\boldsymbol{G}_c. \tag{68}$$

Consequently, $\boldsymbol{G}_i$ necessarily minimize the following optimization problem:

$$\min_{\boldsymbol{Z}\in\mathbb{R}^{a\times\ell}, \boldsymbol{G}_i\in\mathcal{O}(\ell)} \quad \sum_{i=1}^{c} \|\boldsymbol{A}_i\boldsymbol{G}_i - \boldsymbol{Z}\|_{\mathrm{F}}^2. \tag{69}$$

Importantly, Problem (69) involves a matrix norm minimization objective, which inherently exhibits invariance under arbitrary common orthogonal transformations from the right. Specifically, let $\boldsymbol{O} \in \mathcal{O}(\ell)$ be an arbitrary orthogonal matrix. Then, choosing $\boldsymbol{Z}^* = \boldsymbol{A}_1\boldsymbol{O}$ and $\boldsymbol{G}_i^* = \boldsymbol{E}_i^\top\boldsymbol{O}$ yields global minimizers for Problem (69), because:

$$\begin{aligned}\sum_{i=1}^{c} \|\boldsymbol{A}_i\boldsymbol{E}_i^\top\boldsymbol{O} - \boldsymbol{A}_1\boldsymbol{O}\|_{\mathrm{F}}^2 &= \sum_{i=1}^{c} \|\boldsymbol{A}\boldsymbol{F}_i\boldsymbol{E}_i^\top\boldsymbol{O} - \boldsymbol{A}\boldsymbol{F}_1\boldsymbol{O}\|_{\mathrm{F}}^2 \\ &= \sum_{i=1}^{c} \|\boldsymbol{A}\boldsymbol{F}_1\boldsymbol{O} - \boldsymbol{A}\boldsymbol{F}_1\boldsymbol{O}\|_{\mathrm{F}}^2 \\ &= 0.\end{aligned}$$

Given that the analyst has access to $\boldsymbol{A}_i$ for all $i \in [c]$, we may fix $\boldsymbol{Z} = \boldsymbol{A}_1\boldsymbol{O}$ in Problem (69), leading directly to the classical *Orthogonal Procrustes Problem (OPP)*:

$$\min_{\boldsymbol{G}_i\in\mathcal{O}(\ell)} \quad \sum_{i=1}^{c} \|\boldsymbol{A}_i\boldsymbol{G}_i - \boldsymbol{A}_1\boldsymbol{O}\|_{\mathrm{F}}^2. \tag{OPP}$$

The closed-form solutions are given by:

$$\boldsymbol{G}_i^* = \boldsymbol{U}_i\boldsymbol{V}_i^\top, \tag{70}$$

where:

$$\boldsymbol{A}_i^\top\boldsymbol{A}_1\boldsymbol{O} = \boldsymbol{U}_i\boldsymbol{\Sigma}_i\boldsymbol{V}_i^\top, \tag{71}$$

as established in [14].

ODC's basis alignment procedure is summarized in **Algorithm** 4.

Algorithm 4: ODC's Basis Alignment Procedure

Input : $\boldsymbol{A}_i \in \mathbb{R}^{a\times\ell}$ for each user $i \in [c]$
Output: $\boldsymbol{G}_i^* \in \mathcal{O}(\ell)$ for each user $i \in [c]$

begin
  Randomly sample $\boldsymbol{O} \in \mathcal{O}(\ell)$ and compute $\boldsymbol{A}_i^\top \boldsymbol{A}_1 \boldsymbol{O} = \boldsymbol{U}_i \boldsymbol{\Sigma}_i \boldsymbol{V}_i^\top$
  Set $\boldsymbol{G}_i^* := \boldsymbol{U}_i \boldsymbol{V}_i^\top$
end

5.2. Orthogonal Concordance of ODC

A fundamental limitation of weak concordance is its invariance under arbitrary right-multiplication by an invertible matrix. Specifically, if $\boldsymbol{G}_i^*$ satisfy weak concordance, then so do the matrices $\boldsymbol{G}_i^* \boldsymbol{R}$ for any $\boldsymbol{R} \in \mathcal{GL}(\ell)$. In practical settings, however, an ill-conditioned choice of $\boldsymbol{R}$ can severely impair downstream model performance, even though the resulting matrices remain weakly concordant. To eliminate this degree of freedom, we introduce a stricter requirement—*orthogonal concordance*.

Definition 5.2. *(Orthogonal Concordance)*
The orthogonal change-of-basis matrices $\boldsymbol{G}_i \in \mathcal{O}(\ell)$, for $i \in [c]$, satisfy orthogonal concordance if

$$\boldsymbol{F}_1 \boldsymbol{G}_1 = \boldsymbol{F}_2 \boldsymbol{G}_2 = \cdots = \boldsymbol{F}_c \boldsymbol{G}_c. \tag{72}$$

Orthogonal concordance enforces the same alignment condition as weak concordance but additionally constrains each $\boldsymbol{G}_i$ to be orthogonal. Consequently, concordance is preserved only under right-multiplication by orthogonal matrices, not by arbitrary invertible transformations. Because orthogonal transformations are Euclidean isometries, distance-based models such as SVMs are theoretically invariant to such transformations [38]. Moreover, even models that are not strictly distance-based (e.g., MLPs) are empirically observed to exhibit only limited sensitivity, as demonstrated in our experiments (see § 8).

Theorem 5.3. *(Orthogonal Concordance of ODC)*
*Suppose that we observe matrices $\boldsymbol{A}_i = \boldsymbol{A}\boldsymbol{F}_i$, $i \in [c]$, with $\boldsymbol{A} \in \mathbb{R}^{a\times m}$ and $\boldsymbol{F}_i \in \mathbb{R}^{m\times\ell}$. For any orthogonal matrix $\boldsymbol{O} \in \mathcal{O}(\ell)$, let $\boldsymbol{A}_i^\top \boldsymbol{A}_1 \boldsymbol{O} = \boldsymbol{U}_i \boldsymbol{\Sigma}_i \boldsymbol{V}_i^\top$ denote the SVD. Under **Assumption** 5.1, for each $i \in [c]$, the solution $\boldsymbol{G}_i^* = \boldsymbol{U}_i \boldsymbol{V}_i^\top$ to the Orthogonal Procrustes Problem:*

$$\min_{\boldsymbol{G}_i \in \mathcal{O}(\ell)} \quad \sum_{i=1}^{c} \|\boldsymbol{A}_i \boldsymbol{G}_i - \boldsymbol{A}_1 \boldsymbol{O}\|_{\mathrm{F}}^2, \tag{73}$$

satisfies

$$\boldsymbol{G}_i^* = \boldsymbol{E}_i^\top \boldsymbol{O}, \tag{74}$$

*thereby guaranteeing orthogonal concordance (**Definition** 5.2).*

Proof. We prove for all $i \in [c]$. Given (OPP), we can write:

$$\begin{aligned} \|\boldsymbol{A}_i \boldsymbol{G}_i - \boldsymbol{A}_1 \boldsymbol{O}\|_{\mathrm{F}}^2 &= \operatorname{tr}\left((\boldsymbol{A}_i \boldsymbol{G}_i - \boldsymbol{A}_1 \boldsymbol{O})^\top (\boldsymbol{A}_i \boldsymbol{G}_i - \boldsymbol{A}_1 \boldsymbol{O})\right) \\ &= \|\boldsymbol{A}_i\|_{\mathrm{F}}^2 + \|\boldsymbol{A}_1\|_{\mathrm{F}}^2 - 2\operatorname{tr}(\boldsymbol{G}_i^\top \boldsymbol{A}_i^\top \boldsymbol{A}_1 \boldsymbol{O}), \end{aligned}$$

where $\operatorname{tr}(\cdot)$ denotes the matrix trace. Minimizing $\|\boldsymbol{A}_i \boldsymbol{G}_i - \boldsymbol{A}_1 \boldsymbol{O}\|_{\mathrm{F}}^2$ for each $\boldsymbol{G}_i$ individually is equivalent to minimizing $\sum_{i=1}^{c} \|\boldsymbol{A}_i \boldsymbol{G}_i - \boldsymbol{A}_1 \boldsymbol{O}\|_{\mathrm{F}}^2$ for all $\boldsymbol{G}_i$. Therefore, solving (OPP) is equivalent to solving:

$$\max_{\boldsymbol{G}_i \in \mathcal{O}(\ell)} \quad \operatorname{tr}(\boldsymbol{G}_i^\top \boldsymbol{A}_i^\top \boldsymbol{A}_1 \boldsymbol{O}), \tag{75}$$

for each $i \in [c]$.

From condition 3) of **Assumption** 5.1, we have:

$$\boldsymbol{A}_1 \boldsymbol{O} = \boldsymbol{A}\boldsymbol{F}_1 \boldsymbol{O} = \boldsymbol{A}\boldsymbol{F}_i \boldsymbol{E}_i^\top \boldsymbol{O} = \boldsymbol{A}_i \boldsymbol{E}_i^\top \boldsymbol{O}. \tag{76}$$

Substitute this into Problem (75), and let $\boldsymbol{A}_i^\top \boldsymbol{A}_i = \boldsymbol{Q}_i \boldsymbol{\Lambda}_i \boldsymbol{Q}_i^\top$ be the eigenvalue decomposition. We have:

$$\begin{aligned}
\operatorname{tr}(\boldsymbol{G}_i^\top \boldsymbol{A}_i^\top \boldsymbol{A}_1 \boldsymbol{O}) &= \operatorname{tr}(\boldsymbol{G}_i^\top \boldsymbol{A}_i^\top \boldsymbol{A}_i \boldsymbol{E}_i^\top \boldsymbol{O}) \\
&= \operatorname{tr}(\boldsymbol{G}_i^\top \boldsymbol{Q}_i \boldsymbol{\Lambda}_i \boldsymbol{Q}_i^\top \boldsymbol{E}_i^\top \boldsymbol{O}) \\
&= \operatorname{tr}(\boldsymbol{Q}_i^\top \boldsymbol{E}_i^\top \boldsymbol{O} \boldsymbol{G}_i^\top \boldsymbol{Q}_i \boldsymbol{\Lambda}_i) \\
&= \operatorname{tr}(\boldsymbol{W}_i' \boldsymbol{\Lambda}_i) \\
&= \sum_{s=1}^{\ell} w'_{i,(s,s)} \lambda_{i,(s,s)}, \qquad (77)
\end{aligned}$$

where $\boldsymbol{W}_i' = \boldsymbol{Q}_i^\top \boldsymbol{E}_i^\top \boldsymbol{O} \boldsymbol{G}_i^\top \boldsymbol{Q}_i$, and $w'_{i,(s,t)}, \lambda_{i,(s,t)}$ denote the (s,t)-th elements of matrices $\boldsymbol{W}_i'$ and $\boldsymbol{\Lambda}_i$, respectively. Since $\boldsymbol{W}_i' \in O(\ell)$, $w'_{i,(s,t)} \leq 1$ for all s, t. Thus, the sum in (77) is maximized when $\boldsymbol{W}_i' = \boldsymbol{I}$, which gives:

$$\begin{aligned}
\boldsymbol{G}_i^* &= \boldsymbol{Q}_i \boldsymbol{Q}_i^\top \boldsymbol{E}_i^\top \boldsymbol{O} \\
\boldsymbol{G}_i^* &= \boldsymbol{E}_i^\top \boldsymbol{O},
\end{aligned}$$

and therefore, we have

$$\boldsymbol{F}_1 \boldsymbol{G}_1^* = \cdots = \boldsymbol{F}_c \boldsymbol{G}_c^*, \qquad (78)$$

which completes the proof. □

Theorem 5.3 shows that the orthogonal right factor $\boldsymbol{O} \in O(\ell)$ can be chosen arbitrarily while preserving orthogonal concordance. This flexibility naturally raises the question of whether such arbitrary choices might adversely affect downstream model performance. We argue that when assumptions are satisfied, any resulting performance variation is negligible, even for non-distance-based models, and we empirically validate this claim in §8.

5.3. FLOPs of ODC

In **Algorithm 4**, ODC aligns the user-specific bases by first sampling a random orthogonal matrix $\boldsymbol{O} \in O(\ell)$ and then, for each user $i \in [c]$, forming the product $\boldsymbol{A}_i^\top \boldsymbol{A}_1 \boldsymbol{O} \in \mathbb{R}^{\ell \times \ell}$ and computing its singular value decomposition. Using the FLOP counts from Table 2, we can estimate the cost as follows.

For each user i:

- Forming $\boldsymbol{A}_i^\top \boldsymbol{A}_1$ multiplies an $\ell \times a$ matrix by an $a \times \ell$ matrix and thus costs $2a\ell^2$ FLOPs.
- Multiplying by $\boldsymbol{O}$, i.e. computing $\boldsymbol{A}_i^\top \boldsymbol{A}_1 \boldsymbol{O}$, multiplies two $\ell \times \ell$ matrices and costs $2\ell^3$ FLOPs.
- Computing the SVD $\boldsymbol{A}_i^\top \boldsymbol{A}_1 \boldsymbol{O} = \boldsymbol{U}_i \boldsymbol{\Sigma}_i \boldsymbol{V}_i^\top$ of an $\ell \times \ell$ matrix costs approximately $4\ell^3 + 8\ell^3 = 12\ell^3$ FLOPs.
- Forming the alignment matrix $\boldsymbol{G}_i^* = \boldsymbol{U}_i \boldsymbol{V}_i^\top$ is an $\ell \times \ell$ matrix product and costs $2\ell^3$ FLOPs.

Thus, the per-user cost is approximately.

$$2a\ell^2 + 2\ell^3 + 12\ell^3 + 2\ell^3 \;=\; 2a\ell^2 + 16\ell^3 \quad \text{FLOPs.} \qquad (79)$$

Summing over all c users and ignoring the one-time cost of sampling $\boldsymbol{O}$, the total FLOP count for ODC is

$$c(2a\ell^2 + 16\ell^3) \;=\; 2ca\ell^2 + 16c\ell^3 \text{ FLOPs.} \qquad (80)$$

In big-O notation, this can be summarized as

$$O(ca\ell^2 + c\ell^3) = O(ac\ell^2). \qquad (81)$$

Table 4: Computational Time Complexity Comparison of Basis-Alignment

	Imakura-DC	**Kawakami-DC**	**ODC**
Computational Time Complexity	$O\left(\min\{a(c\ell)^2,\, a^2c\ell\}\right)$	$O\left(\min\{a(c\ell)^2,\, a^2c\ell\}\right)$	$O(ac\ell^2)$

5.4. Peak memory of ODC

We again measure memory in units of real scalars. The c matrices $\boldsymbol{A}_i$ are stored explicitly, requiring

$$\sum_{i=1}^{c} a\ell = ac\ell \tag{82}$$

scalars in total. During ODC, we also store the random orthogonal matrix $\boldsymbol{O} \in O(\ell)$, which costs ℓ^2 scalars, and the alignment matrices $\boldsymbol{G}_i^* \subset \mathbb{R}^{\ell\times\ell}$, which cost $c\ell^2$ scalars once all users have been processed.

For a fixed user $i \in [c]$, the computation of $\boldsymbol{A}_i^\top \boldsymbol{A}_1 \boldsymbol{O}$ and its SVD can be organized so that only $O(\ell^2)$ additional working storage is needed:

- We form an intermediate product $\boldsymbol{B}_i = \boldsymbol{A}_i^\top \boldsymbol{A}_1 \in \mathbb{R}^{\ell\times\ell}$ (or equivalently $\boldsymbol{B}_i = \boldsymbol{A}_i^\top(\boldsymbol{A}_1\boldsymbol{O})$) and overwrite it with $\boldsymbol{B}_i\boldsymbol{O}$. This requires a single $\ell \times \ell$ buffer, i.e. ℓ^2 scalars.
- Computing the SVD $\boldsymbol{B}_i = \boldsymbol{U}_i\boldsymbol{\Sigma}_i\boldsymbol{V}_i^\top$ for an $\ell \times \ell$ matrix requires storing $\boldsymbol{U}_i, \boldsymbol{V}_i \in \mathbb{R}^{\ell\times\ell}$, the diagonal $\boldsymbol{\Sigma}_i$, and SVD workspace, all together costing $O(\ell^2)$ scalars.
- Forming the alignment matrix $\boldsymbol{G}_i^* = \boldsymbol{U}_i\boldsymbol{V}_i^\top \in \mathbb{R}^{\ell\times\ell}$ requires another ℓ^2 scalars, after which $\boldsymbol{U}_i$ and $\boldsymbol{V}_i$ can be discarded. Thus, at any given time, the per-user working storage is $O(\ell^2)$.

Near the end of the loop over users, we therefore store all inputs $\boldsymbol{A}_i$ ($ac\ell$ scalars), all outputs $\boldsymbol{G}_i^*$ ($c\ell^2$ scalars), the random orthogonal matrix $\boldsymbol{O}$ (ℓ^2 scalars), and $O(\ell^2)$ scalars of temporary workspace. Hence, the peak memory requirement of ODC is

$$ac\ell + c\ell^2 + O(\ell^2). \tag{83}$$

Under the natural regime $a > \ell$, this is dominated by the storage of the transformed anchor data, and we also have

$$\Theta(ac\ell). \tag{84}$$

6. Empirics on Time Efficiency

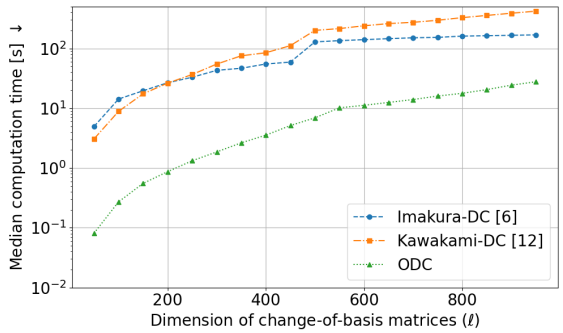
The computational time complexities (in big-O notation) of all basis-alignment methods are summarized in Table 4. As shown in Table 4, under the natural dimensional ordering $\ell \le m < a$, the computational time complexity of ODC satisfies

$$ac\ell^2 \;<\; \min\{a(c\ell)^2,\, a^2c\ell\}. \tag{85}$$

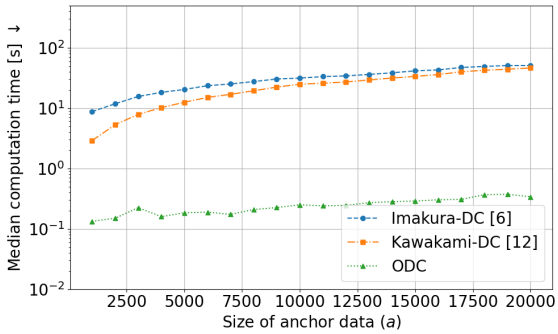
This implies that ODC inherently incurs a lower computational cost than both Imakura-DC and Kawakami-DC. To empirically corroborate this theoretical advantage, we measured wall-clock execution times under controlled conditions.

All three methods admit closed-form expressions for the change-of-basis matrices $\boldsymbol{G}_i$. Consequently, runtime depends only on the dimensions of the local anchor representations $\boldsymbol{A}_i \in \mathbb{R}^{a\times\ell}$ for $i \in [c]$, rather than on $\boldsymbol{X}_i$ or n_i. Because the global anchor $\boldsymbol{A} \in \mathbb{R}^{a\times m}$ is sampled i.i.d. from the uniform distribution over $[0, 1)$ independently of the private datasets $\boldsymbol{X}_i \in \mathbb{R}^{n_i\times m}$, the alignment cost is isolated from user-specific data characteristics.

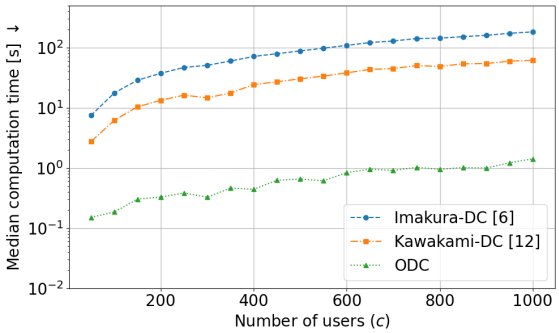
For the empirical evaluation, we generated uniformly random matrices $\boldsymbol{A}_i \in \mathbb{R}^{a\times\ell}$, $i \in [c]$, and varied each of the three primary parameters (anchor size a, latent dimension ℓ, and the number of users c), while keeping the other two fixed. The specific experimental settings are summarized in Table 5. As prior literature [12] suggests, randomized SVD improves computational efficiency for contemporary DC methods; thus, we employed it for both Imakura-DC and Kawakami-DC. Conversely, ODC remained unchanged, as it inherently requires only a single full SVD of an



(a) Wall-clock time versus latent dimension $\ell\,(a, c) = (1000, 50)$.



(b) Wall-clock time versus anchor size $a\,(\ell, c) = (50, 50)$.



(c) Wall-clock time versus number of users $c\,(a, \ell) = (1000, 50)$.

Figure 6: Wall-clock time with varying parameters (ℓ, a, c).

$\ell \times \ell$ matrix. Each experiment was repeated 100 times, and we report median (instead of mean) runtimes to mitigate transient system effects.

All CPU experiments were run on Google Colab Pro+ (CPU, high-memory runtime). The virtual machine reported Linux 6.6.105+ (x86_64, glibc 2.35) as the operating system, an Intel(R) Xeon(R) CPU @ 2.20 GHz with 8 logical cores (2 threads per core), and approximately 51 GiB of RAM. We used Python 3.12.12, NumPy 2.0.2, and SciPy 1.16.3. Linear algebra kernels were provided by the `scipy-openblas` build of OpenBLAS 0.3.27 (64-bit integer interface, `DYNAMIC_ARCH`, pthreads threading layer, Haswell configuration). We pinned threads via environment variables `OPENBLAS_NUM_THREADS=4`, `OMP_NUM_THREADS=4`, and `MKL_NUM_THREADS=4` (where applicable), and used double-precision (float64) C-contiguous arrays throughout.

Figure 6 plots the measured median running times. Across all parameter variations, ODC consistently outperforms contemporary DC approaches, with empirical speed-ups ranging from roughly 6× to over 100×, validating the theoretical complexity hierarchy detailed in Table 4.

Table 5: Design of the efficiency experiment. Two parameters are held fixed while the third is swept over the indicated range.

Figure	**Free parameter (fixed pair)**	**Range**
Fig. 6(a)	dimension ℓ $(a, c) = (1000, 50)$	50:50:950
Fig. 6(b)	anchor size a $(\ell, c) = (50, 50)$	1000:1000:20000
Fig. 6(c)	users c $(a, \ell) = (1000, 50)$	50:50:1000

6.1. Scaling with Latent Dimension ℓ

Fig. 6(a) indicates an approximate power-law scaling of computation time with respect to the dimension ℓ. We applied ordinary least squares (OLS) regression to the linear model:

$$\log_{10}(\text{time}) = \kappa + \alpha \log_{10}(\ell), \tag{86}$$

with the constant κ to empirically estimate the exponent α. The resulting empirical estimates $\widehat{\alpha}_\ell$ are:

$$\widehat{\alpha}_\ell = \begin{cases} 1.26, & \text{Imakura–DC} \\ 1.75, & \text{Kawakami–DC} \\ 2.05, & \text{ODC.} \end{cases} \tag{87}$$

For these log–log fits, the coefficients of determination are $R^2 \approx 0.97$, 0.99, and 0.99 for Imakura–DC, Kawakami–DC, and ODC, respectively. The residuals in $\log_{10}(\text{time})$ are small, show no discernible trend with $\log_{10}(\ell)$, and are approximately symmetric about zero, indicating that a simple power-law model in ℓ is adequate over the explored range.

The observed exponent $\alpha \approx 2$ for ODC aligns closely with its theoretically derived complexity $O(ac\ell^2)$. Conversely, contemporary DC methods exhibit somewhat smaller empirical exponents, reflecting complexity consistent with their theoretical predictions of $O(\min\{a(c\ell)^2, a^2c\ell\})$. Although ODC has a slightly higher exponent with respect to ℓ, its absolute computational cost remains substantially lower across the tested range of dimensions. This practical efficiency arises primarily from ODC avoiding explicit construction and manipulation of the large, dense $a \times c\ell$ concatenated matrix, thereby circumventing costly large-scale singular value decompositions. Moreover, typical practical scenarios involve $a \gg \ell$; thus, the substantial scaling advantage in a (discussed further in § 6.2) renders the modestly larger exponent in ℓ negligible in realistic deployment.

6.2. Scaling with Anchor Size a

In Fig. 6(b), the relationship between computational wall-clock time and anchor size a again exhibits an approximate power-law scaling. Applying OLS regression to the log-log relationship $\log_{10}(\text{time}) = \kappa + \alpha \log_{10}(a)$ yields the following empirical estimates $\widehat{\alpha}_a$:

$$\widehat{\alpha}_a = \begin{cases} 0.79 & \text{Imakura–DC,} \\ 0.92 & \text{Kawakami–DC,} \\ 0.56 & \text{ODC.} \end{cases} \tag{88}$$

The power-law model in a again provides a good description of the data, with $R^2 \approx 0.99$, 0.998, and 0.87 for Imakura–DC, Kawakami–DC, and ODC, respectively. Residuals in $\log_{10}(\text{time})$ show no clear heteroscedasticity or trend with $\log_{10}(a)$; for the two baselines they are tightly concentrated around zero, while for ODC they are somewhat more dispersed but remain roughly symmetric, indicating that deviations from an exact power law are modest.

Notably, all empirically observed slopes are lower than their corresponding theoretical predictions ($\alpha = 1$ for ODC and $\alpha \in \{1, 2\}$ for baseline DC methods). The significantly lower slope of 0.56 observed for ODC is particularly noteworthy. This empirical observation can be attributed to ODC's dominant computational workload comprising matrix multiplications—specifically, performing c independent multiplications of relatively small $\ell \times a$ and $a \times \ell$ matrices—which are highly optimized and efficiently executed in modern computational environments, thus performing substantially better in practice than theoretically anticipated.

At the maximum tested anchor size $a = 20\,000$, median wall-clock runtimes are approximately 48.5 s for Imakura–DC, 50.4 s for Kawakami–DC, and 0.47 s for ODC, corresponding to empirical speed-up factors of about 104× and 108× in favor of the proposed ODC method—more than two orders of magnitude in this regime.

6.3. Scaling with the Number of Users c

Fig. 6(c) examines computational scaling with respect to the number of users c, holding the dimensions $(a, \ell) = (1000, 50)$ fixed. Applying OLS regression to the log-log relationship $\log_{10}(\text{time}) = \kappa + \alpha \log_{10}(c)$, we obtain the following empirical estimates $\widehat{\alpha}_c$:

$$\widehat{\alpha}_c = \begin{cases} 0.95 & \text{Imakura–DC,} \\ 0.92 & \text{Kawakami–DC,} \\ 0.84 & \text{ODC.} \end{cases} \tag{89}$$

These log–log regressions also exhibit high goodness-of-fit, with $R^2 \approx 0.99$, 0.98, and 0.94 for Imakura–DC, Kawakami–DC, and ODC, respectively. The residuals in $\log_{10}(\text{time})$ are small in magnitude, roughly homoscedastic in $\log_{10}(c)$, and display only mild asymmetry, suggesting that a linear power-law dependence on c is an adequate description over the explored range.

For ODC, the theoretical complexity is $O(ac\ell^2)$, predicting a slope $\alpha \approx 1$. The slightly lower empirical slope of 0.84 is primarily due to efficient parallelization and vectorization in modern CPU architectures. Conversely, the theoretical complexity for Imakura–DC and Kawakami–DC methods is $O(\min\{a(c\ell)^2, a^2c\ell\})$. Under the tested dimensions, the $a^2c\ell$ term dominates for $c > 20$, predicting linear scaling in c. The empirical slopes of 0.95 and 0.92 closely align with these theoretical predictions.

To connect this scaling to the user-facing notion of *incremental user latency*, we further analyzed the same experiment on a linear scale by regressing wall-clock time $T(c)$ against c via

$$T(c) = \beta_0 + \beta_1 c, \tag{90}$$

where β_1 represents the average increase in computation time per additional user (seconds/user) on our hardware. For $(a, \ell) = (1000, 50)$, this yields:

$$\beta_1 \approx \begin{cases} 8.99 \times 10^{-2} \text{ s/user} & \text{Imakura–DC,} \\ 4.76 \times 10^{-2} \text{ s/user} & \text{Kawakami–DC,} \\ 8.67 \times 10^{-4} \text{ s/user} & \text{ODC,} \end{cases} \tag{91}$$

with 95% confidence intervals $[0.084, 0.096]$, $[0.041, 0.054]$, and $[0.00066, 0.00107]$ s/user, respectively, and coefficients of determination $R^2 \approx 0.98$, 0.94, and 0.83. Residuals from these linear fits are unsystematic in c and small relative to the fitted trend, so over the investigated range $T(c)$ is well-approximated by an affine function of c.

In other words, in this setting, ODC incurs only about 0.00087 seconds (0.87 ms) of additional computation per extra user, compared to 0.090 and 0.048 seconds for Imakura–DC and Kawakami–DC. Thus, ODC achieves roughly 100× lower incremental user latency than Imakura–DC and 55× lower than Kawakami–DC. Over blocks of 50 additional users, these slopes correspond to an extra 4.5 s (Imakura–DC), 2.4 s (Kawakami–DC), and only 0.043 s (ODC).

Although asymptotic scaling behaviors are similar, ODC exhibits significantly improved absolute performance. Median runtimes at $c = 1000$ report that ODC completes computations in only 1.0 s, compared to approximately 98 s and 53 s required by Imakura–DC and Kawakami–DC, respectively. Thus, ODC achieves empirical speed-ups of at least 50×.

This substantial performance gap arises because ODC employs optimized batched matrix multiplications and small-scale $(\ell \times \ell)$ singular value decompositions, while baseline methods incrementally extend randomized SVD computations with each additional user. In practice, ODC maintains nearly constant incremental computational overhead per user, rendering the basis-alignment phase computationally negligible in realistic deployments.

7. Empirics Under Relaxed Assumptions

Our theoretical analysis in § 5 relies on two key assumptions regarding the secret bases $\boldsymbol{F}_i$:

1. **Identical Span**: all $\boldsymbol{F}_i$ share an identical column space (**Assumption** 5.1-2);
2. **Orthonormality**: each $\boldsymbol{F}_i$ has orthonormal columns (**Assumption** 5.1-3).

Table 6: Summary of secret-basis conditions evaluated.

Condition	Identical Span	Orthonormality
SameSpan-Orth	✓	✓
SameSpan	✓	×
DiffSpan-Orth	×	✓
DiffSpan	×	×

In practical scenarios, these ideal conditions may not strictly hold, particularly when bases are generated through computationally inexpensive random projections or derived from heterogeneous local datasets. We thus empirically quantify the impact of deviations from these assumptions on performance.

7.1. Experimental Design

We evaluate four controlled scenarios with varying adherence to these assumptions (Table 6). Let $\boldsymbol{V}_i \in \mathbb{R}^{m\times\ell}$ denote the top ℓ right singular vectors of $\boldsymbol{X}_i$.

For each client i, we instantiate the secret basis $\boldsymbol{F}_i$ as

$$\boldsymbol{F}_i \;=\; \begin{cases} \boldsymbol{V}_1\,\boldsymbol{E}_i, & \textbf{SameSpan-Orth}:\ \boldsymbol{E}_i \in \mathcal{O}(\ell)\ \text{(Haar)};\\ \boldsymbol{V}_1\,\boldsymbol{E}_i, & \textbf{SameSpan}:\ \boldsymbol{E}_i \sim \mathrm{Unif}([0,1))^{\ell\times\ell};\\ \boldsymbol{V}_i\,\boldsymbol{E}_i, & \textbf{DiffSpan-Orth}:\ \boldsymbol{E}_i \in \mathcal{O}(\ell)\ \text{(Haar)};\\ \boldsymbol{V}_i\,\boldsymbol{E}_i, & \textbf{DiffSpan}:\ \boldsymbol{E}_i \sim \mathrm{Unif}([0,1))^{\ell\times\ell}, \end{cases} \tag{92}$$

where $\mathrm{Unif}([0,1))^{\ell\times\ell}$ denotes an $\ell\times\ell$ matrix with i.i.d. entries drawn from $[0,1)$. Thus, **SameSpan** fixes $\mathrm{span}(\boldsymbol{F}_i) = \mathrm{span}(\boldsymbol{V}_1)$ for all users, whereas **DiffSpan** allows client-specific spans $\mathrm{span}(\boldsymbol{F}_i) = \mathrm{span}(\boldsymbol{V}_i)$; **Orthnormality** is controlled by sampling $\boldsymbol{E}_i$ either Haar-uniformly from $\mathcal{O}(\ell)$ or entrywise from $\mathrm{Unif}([0,1))$.

We group tasks into five domains:

- **Image classification (MNIST, Fashion-MNIST):** Images are flattened into 28×28 vectors and normalized to $[0,1]$. We randomly select 10,000 samples for the test set. We report mean classification accuracy for SVM and MLP classifiers, averaged over 100 independent runs. The anchor matrix is drawn uniformly at random with $a = 1000$.
- **Biomedical compound classification (TDC [39]):** Molecules are featurized as 2048-bit Morgan fingerprints (radius = 2). Data are split across four users, with each user holding samples from exactly one class (see Fig. 7). We use the dataset-provided train/test partitions and evaluate performance using ROC-AUC and PR-AUC (MLP), averaged over 100 runs. The anchor matrix is drawn uniformly at random with $a = 3000$.
- **Income classification (Adult [40]):** We use the Adult income dataset and partition it across 200 users using both horizontal and *vertical* splits. We randomly select 10,000 samples for the test set. Under the vertical split, half of the users possess one subset of features, while the remaining users possess the complementary subset. We report the mean classification accuracy of an MLP classifier over 100 runs. The anchor matrix is drawn uniformly at random with $a = 1000$.
- **Facial attribute classification (CelebA [24]):** RGB images (128×128) are flattened and normalized to $[0,1]$. In each iteration, we randomly sample 4000 images from the full dataset and select 1000 for testing. We measure binary gender prediction accuracy using an MLP classifier, averaged over 100 runs, and additionally compare against (ε,δ)-DP baselines. The anchor matrix is drawn uniformly at random with $a = 10000$.
- **Clinical regression (eICU-CRD [41]):** We use data from the 50 largest hospitals to predict patient length of stay (days) from 25 routinely collected clinical features. We randomly select 20% of the data as the test set. Performance is reported as RMSE (lower is better), averaged across hospitals over 100 runs using an MLP model. As a federated learning baseline, we run federated averaging (FedAvg) for 40 rounds with full participation. The anchor matrix is drawn uniformly at random with $a = 3000$.

We compare a centralized oracle (**Central**); per-user local models (**Local**); two contemporary DC baselines **Imakura-DC** (**Algorithm** 2), **Kawakami-DC** (**Algorithm** 3); and the proposed **ODC** (**Algorithm** 4). For CelebA, we additionally include additive Gaussian **DP** [42], and for eICU, we include **FedAvg** [2], to place ODC in the wider PPML context.

The downstream models are configured as follows. For the SVM baseline, we use an RBF kernel with $C = 1.0$ and γ = "scale". The MLP baseline consists of a single hidden layer with 256 ReLU units, trained with Adam (`solver = "adam"`) using a mini-batch size of 32 and a maximum of 1000 training iterations, with early stopping enabled.

All other experimental settings (e.g., hardware, software, and implementation details) are identical to those described in § 6, unless otherwise stated.

All numerical results reported in this paper, including all baselines, were produced by the authors by executing each method on the *same* dataset partitions described above. We do not copy performance numbers from prior publications. To ensure fair comparisons, we keep the latent dimension ℓ and anchor size a fixed across DC/ODC variants, and we use the same downstream-model hyperparameters described above for all compared methods.

7.2. Results and Discussion

Table 7: Performance comparison of ODC and baseline DC methods across multiple datasets (MNIST, Fashion-MNIST, and biomedical TDC datasets) using various classifiers and performance metrics. The reported results represent the mean scores (± margin of error at 95% confidence, computed over 100 independent trials) under the four distinct secret-base conditions described in Table 6.

Dataset	Secret Bases	SVM Accuracy [%] ↑					MLP Accuracy [%] ↑				
		Central	**Local**	**Imakura-DC [6]**	**Kawakami-DC [12]**	**ODC**	**Central**	**Local**	**Imakura-DC [6]**	**Kawakami-DC [12]**	**ODC**
MNIST [50]	**SameSpan-Orth**	96.0±0.1	66.0±0.9	**96.1±0.1**	**96.1±0.1**	95.6±0.1	95.1±0.1	59.7±1.4	**96.0±0.1**	95.9±0.1	95.9±0.1
	SameSpan			**96.1±0.1**	**96.1±0.1**	88.8±0.2			**95.9±0.1**	95.8±0.1	92.6±0.2
	DiffSpan-Orth			**95.1±0.1**	94.7±0.1	94.9±0.1			94.4±0.2	93.2±0.2	**95.3±0.1**
	DiffSpan			**94.9±0.1**	94.7±0.1	86.5±0.3			**94.9±0.1**	93.2±0.2	91.2±0.2
Fashion-MNIST [51]	**SameSpan-Orth**	86.1±0.2	62.9±0.8	**85.4±0.2**	**85.4±0.2**	83.7±0.3	86.1±0.2	61.1±1.0	**86.1±0.2**	86.0±0.2	86.0±0.2
	SameSpan			**85.4±0.2**	**85.4±0.2**	77.7±0.3			**86.1±0.2**	85.9±0.2	80.5±0.3
	DiffSpan-Orth			**83.4±0.2**	83.0±0.3	82.5±0.2			81.8±0.3	81.5±0.3	**84.3±0.2**
	DiffSpan			82.9±0.2	**83.0±0.3**	76.3±0.3			**82.5±0.3**	81.5±0.3	78.8±0.4

Dataset	Secret Bases	ROC-AUC [%] ↑				PR-AUC [%] ↑			
		Central	**Imakura-DC [6]**	**Kawakami-DC [12]**	**ODC**	**Central**	**Imakura-DC [6]**	**Kawakami-DC [12]**	**ODC**
AMES [43]	**SameSpan-Orth**	87.1±0.0	86.2±0.4	82.4±0.3	**88.6±0.1**	88.7±0.0	87.9±0.4	83.8±0.4	**89.9±0.1**
	SameSpan		**86.6±0.3**	82.2±0.3	59.9±0.4		**88.2±0.3**	83.7±0.3	62.5±0.4
	DiffSpan-Orth		63.1±0.4	65.4±0.4	**67.9±0.3**		66.9±0.4	69.1±0.4	**71.5±0.3**
	DiffSpan		61.9±0.3	**65.2±0.4**	58.1±0.4		65.6±0.3	**69.0±0.4**	62.0±0.4
Tox21_SR-ARE [44]	**SameSpan-Orth**	75.0±0.0	69.4±1.0	64.5±1.3	**75.4±0.3**	42.2±0.0	30.8±1.1	25.2±1.3	**38.4±0.6**
	SameSpan		**69.0±1.0**	64.5±1.3	54.0±0.3		**30.4±1.1**	25.2±1.2	16.7±0.2
	DiffSpan-Orth		57.4±0.3	58.3±0.3	**61.0±0.3**		19.4±0.2	20.2±0.2	**22.6±0.3**
	DiffSpan		56.6±0.3	**58.5±0.3**	53.9±0.3		18.9±0.2	**20.3±0.2**	17.3±0.2
HIV [45]	**SameSpan-Orth**	78.8±0.0	79.6±0.4	78.1±0.5	**80.8±0.2**	42.0±0.0	41.6±0.5	39.1±0.7	**41.8±0.4**
	SameSpan		**80.1±0.3**	78.0±0.5	58.8±0.4		**42.2±0.4**	38.9±0.8	5.3±0.1
	DiffSpan-Orth		61.1±0.4	60.9±0.4	**67.4±0.3**		7.8±0.2	8.7±0.2	**13.4±0.3**
	DiffSpan		58.1±0.4	**60.8±0.4**	56.3±0.4		6.6±0.2	**8.5±0.3**	4.6±0.1
CYP3A4 [46]	**SameSpan-Orth**	87.2±0.0	84.8±0.2	82.9±0.2	**86.2±0.1**	83.5±0.0	80.6±0.3	78.1±0.3	**82.1±0.1**
	SameSpan		**84.7±0.2**	82.9±0.2	59.2±0.4		**80.5±0.3**	78.1±0.3	49.8±0.5
	DiffSpan-Orth		57.8±0.3	59.4±0.4	**65.5±0.3**		49.8±0.3	51.5±0.5	**57.2±0.3**
	DiffSpan		58.3±0.3	**59.1±0.4**	58.2±0.3		50.4±0.3	**51.1±0.5**	48.9±0.4
CYP2D6 [46]	**SameSpan-Orth**	83.5±0.0	81.4±0.3	80.4±0.2	**83.3±0.1**	63.5±0.0	60.2±0.4	58.5±0.3	**62.3±0.2**
	SameSpan		**81.8±0.2**	80.3±0.2	59.5±0.4		**60.6±0.4**	58.3±0.3	24.2±0.3
	DiffSpan-Orth		62.4±0.4	64.6±0.4	**65.8±0.2**		27.5±0.5	30.3±0.5	**31.2±0.3**
	DiffSpan		62.6±0.3	**64.7±0.4**	57.1±0.4		27.9±0.4	**30.4±0.5**	23.0±0.3
CYP1A2 [46]	**SameSpan-Orth**	91.2±0.0	89.7±0.1	89.5±0.2	**90.4±0.2**	90.1±0.0	88.8±0.1	88.4±0.3	**89.8±0.2**
	SameSpan		**90.3±0.1**	89.2±0.2	65.1±0.5		**89.4±0.1**	88.1±0.3	59.4±0.4
	DiffSpan-Orth		67.3±0.4	**68.6±0.4**	67.9±0.3		62.8±0.5	**64.9±0.5**	63.9±0.3
	DiffSpan		67.8±0.4	**68.7±0.4**	60.2±0.5		63.3±0.4	**64.9±0.5**	55.9±0.5

Tables 7–11 comprehensively summarize model performance under four distinct secret-basis scenarios, while Figure 8 provides complementary visual evidence of privacy preservation. The following discussion is organized around three comparative perspectives:

I. ODC *vs.* Existing DC Methods Under Various Secret-Basis Conditions (Tables 7 and 8)

Table 8: Performance comparison of ODC and baseline DC methods on the Adult dataset using an MLP classifier. The reported results represent the mean accuracy (± margin of error at 95% confidence, computed over 100 independent trials, higher is better) under the four distinct secret-base conditions described in Table 6.

Secret Bases	Central	Local	Imakura-DC [6]	Kawakami-DC [12]	ODC
SameSpan-Orth	84.9±0.2	74.2±1.0	84.9±0.3	84.8±0.3	**85.0±0.2**
SameSpan			**84.9±0.3**	84.8±0.2	83.0±0.3
DiffSpan-Orth			**84.5±0.3**	84.4±0.3	**84.5±0.3**
DiffSpan			84.3±0.3	**84.4±0.3**	82.4±0.3

Table 9: Comparison of classification accuracy on the CelebA dataset using an MLP classifier across different PPML approaches. The reported results represent the mean accuracy (± margin of error at 95% confidence, computed over 100 independent trials) under the four distinct secret-base conditions described in Table 6. Compared methods include models trained using **DP**-based additive Gaussian noise at three privacy budget levels ($\varepsilon = 0.5, 2, 8$ and $\delta = 10^{-3}$ fixed).

Secret Bases	Central	Local	DP [42] ($\varepsilon = 0.5$)	DP [42] ($\varepsilon = 2$)	DP [42] ($\varepsilon = 8$)	Imakura-DC [6]	Kawakami-DC [12]	ODC
SameSpan-Orth	88.4±0.2	75.7±0.6	69.1±0.5	79.5±0.3	83.8±0.3	**86.5±0.2**	86.2±0.2	86.2±0.2
SameSpan			69.1±0.5	79.5±0.3	83.8±0.3	**86.3±0.2**	85.8±0.3	79.9±0.3
DiffSpan-Orth			69.1±0.5	79.5±0.3	**83.8±0.3**	82.7±0.2	82.8±0.3	83.6±0.3
DiffSpan			69.1±0.5	79.5±0.3	**83.8±0.3**	82.6±0.2	82.8±0.3	78.6±0.3

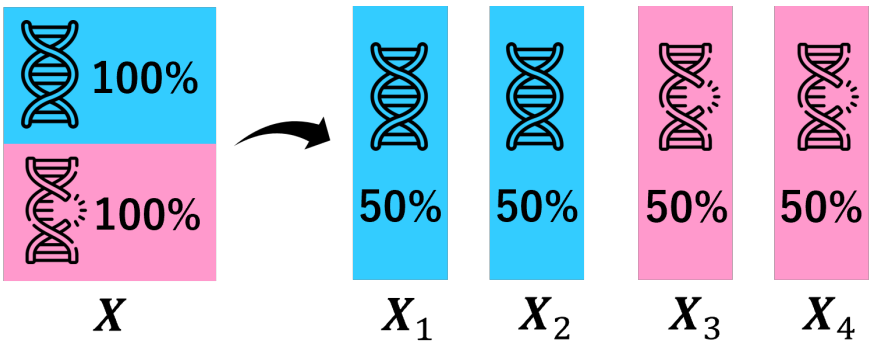
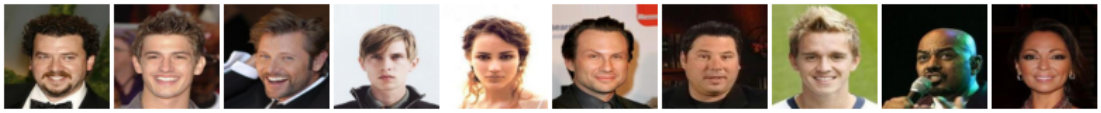
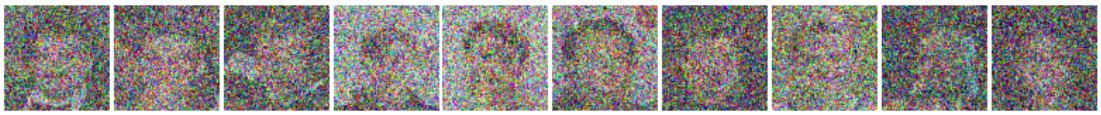


Figure 7: Illustration of extremely heterogeneous splitting applied to TDC datasets. Binary-labeled data are partitioned across four users, each of whom exclusively holds samples of a single label.

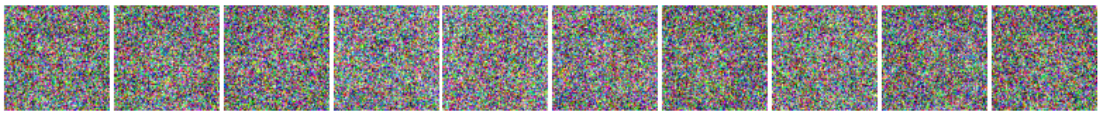


(a) Original images

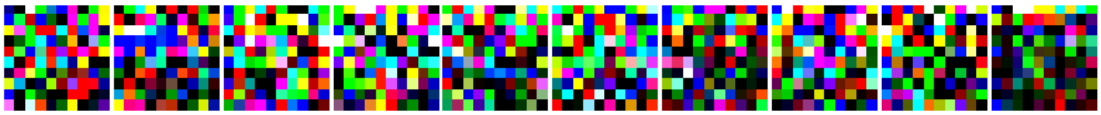
(b) Noise-added images with (ε, δ)-DP, $(\varepsilon, \delta) = (8, 10^{-3})$



(c) Noise-added images with (ε, δ)-DP, $(\varepsilon, \delta) = (2, 10^{-3})$



(d) Noise-added images with (ε, δ)-DP, $(\varepsilon, \delta) = (0.5, 10^{-3})$



(e) Orthogonally projected images ($\ell = 300$)

(f) Randomly projected images ($\ell = 300$)

Figure 8: Visual privacy analysis using CelebA.

Table 10: Face identifiability on CelebA measured by face-verification ROC-AUC using pretrained FaceNet embeddings (higher ⇒ more identifiable; 0.5 = chance).

Representation	Verification AUC ↓
Original	0.977
Noise-added images with (ε, δ)-DP [42], $(\varepsilon, \delta) = (8, 10^{-3})$	0.671
Noise-added images with (ε, δ)-DP [42], $(\varepsilon, \delta) = (2, 10^{-3})$	0.618
Noise-added images with (ε, δ)-DP [42], $(\varepsilon, \delta) = (0.5, 10^{-3})$	0.522
Orthogonally projected images ($\ell = 300$)	0.463
Randomly projected images ($\ell = 300$)	0.485

Table 11: Comparison of RMSE on the eICU-CRD dataset using an MLP classifier across different PPML approaches. The reported results represent the mean RMSE (± margin of error at 95% confidence, computed over 100 independent, lower is better) under the four distinct secret-base conditions described in Table 6. Compared methods include: a **FedAvg** model with full participation.

Secret Bases	**Central**	**Local**	**FedAvg [2]**	**Imakura-DC [6]**	**Kawakami-DC [12]**	**ODC**
SameSpan-Orth	3.996±0.046	4.174±0.048	**4.064±0.049**	4.089±0.049	4.096±0.049	4.083±0.048
SameSpan				4.088±0.049	4.096±0.049	4.154±0.049
DiffSpan-Orth				4.085±0.049	4.090±0.049	4.070±0.048
DiffSpan				4.085±0.049	4.090±0.049	4.150±0.049

- **SameSpan-Orth (identical span and orthonormality satisfied):** Under ideal conditions, ODC achieves performance equal to or marginally better than the centralized oracle across multiple datasets (e.g., MNIST MLP accuracy 95.9 %, matching the oracle; see Table 7). Similar performance from Imakura-DC and Kawakami-DC confirms effective utilization of this ideal scenario by all methods.
- **SameSpan (orthonormality violated):** Dropping the orthonormality constraint significantly impacts ODC's performance—MNIST SVM accuracy drops from 95.6 % to 88.8 % and AMES ROC-AUC from 88.6 % to 59.9 %. Imakura-DC and Kawakami-DC remain largely unaffected, underscoring the theoretical dependence of ODC on orthonormal secret bases.
- **DiffSpan-Orth (orthonormality maintained, identical span violated):** When orthonormality is enforced, ODC demonstrates robustness against subspace misalignment, maintaining accuracy in the range 94.9 %–95.3 % on MNIST, and often surpassing DC baselines in biomedical tasks (e.g., HIV ROC-AUC 67.4 % compared to Imakura-DC 61.1 %; Table 7). Thus, exact span alignment is advantageous but not critical, provided orthonormality holds.
- **DiffSpan (both conditions violated):** Violating both conditions negatively affects all methods, with ODC showing the greatest degradation (e.g., MNIST SVM accuracy drops to 86.5 %). This observation highlights the practical necessity of orthonormal bases, particularly in heterogeneous, high-variance settings.

Regarding Table 8, we observe performance trends mirror those observed under purely horizontal splits, despite the additional vertical data splitting.

II. ODC *vs.* DP-based Perturbation (Table 9)

Figure 8 provides a qualitative visualization, but visual inspection alone can be subjective. We therefore add a quantitative identifiability metric based on an off-the-shelf face-recognition model: we compute 512D FaceNet embeddings [47] (InceptionResnetV1 [48] trained on VGGFace2 [49]) and evaluate face verification using ROC-AUC from cosine similarity on same-identity vs. different-identity pairs. Here, AUC ≈ 1 indicates high identifiability, while AUC ≈ 0.5 corresponds to chance-level identifiability. Table 10 reports this metric for the same privatized representations shown in Fig. 8.

Table 9 summarizes the *utility* side. Under **DiffSpan-Orth** (heterogeneous users with orthonormal secret bases but without an identical-span constraint), ODC reaches 83.6% accuracy, substantially exceeding the stricter DP

settings ($\varepsilon = 0.5$ and $\varepsilon = 2$) and closely approaching DP at $\varepsilon = 8$. In other words, when orthonormality holds, ODC preserves strong predictive performance even outside the idealized identical-span regime.

Taken together, Fig. 8 and Table 10 clarify *why* the DP baselines can look visually weak at larger privacy budgets: across the tested ε values, DP perturbation reduces identifiability but still leaves AUC well above chance ($\approx$ 0.52–0.67), whereas DC-style projections reduce identifiability toward chance (AUC $\approx$ 0.46–0.49). At the same time, Table 9 shows the familiar trade-off: increasing noise for stronger DP-style obfuscation is accompanied by a considerable utility drop, while ODC attains strong obfuscation without requiring a calibrated privacy budget.

Finally, the four secret-base conditions in Table 9 expose an important practical constraint: ODC is visibly more sensitive to violations of orthonormality (**SameSpan** and **DiffSpan**) than the two DC baselines. This aligns with ODC's design goal (orthogonal concordance) and motivates the choice of orthonormal secret bases in practice (e.g., via PCA/SVD/QR), which is typically feasible in DC pipelines.

Table 9 shows that under realistic conditions (DiffSpan–Orth), ODC attains 83.6 % accuracy, surpassing DP at lower privacy budgets by substantial margins (+14.5 pp for $\varepsilon = 0.5$ and +4.1 pp for $\varepsilon = 2$), while nearly matching DP at $\varepsilon = 8$ (only 0.2 pp below). Figure 8 further demonstrates that DP at high privacy budgets preserves visually identifiable features, whereas ODC fully obfuscates visual identity. Thus, ODC provides superior privacy–utility trade-offs without explicitly calibrating privacy budgets.

However, the formal privacy mechanisms underpinning ODC and contemporary DC currently rely on a semi-honest assumption; as a result, these mechanisms remain underdeveloped for scenarios involving stronger or malicious adversaries. Suppose the semi-honest privacy assumption cannot be guaranteed or tolerated. In that case, DP-based additive perturbation may still be preferred, despite the advantageous privacy-performance trade-offs offered by DC methods. This preference arises from the rigorous privacy guarantees inherent in DP.

III. ODC *vs.* Federated Learning (Table 11)

Table 11 shows FedAvg achieving the lowest RMSE (4.060). ODC, under DiffSpan–Orth, is statistically indistinguishable (4.065, 0.005 difference), clearly outperforming Imakura-DC and Kawakami-DC (~4.086). Given FedAvg's requirement for iterative communication, ODC offers a strong alternative, achieving near-FedAvg performance with significantly reduced communication overhead. Interestingly, enforcing identical span conditions negatively affects performance for eICU-CRD, indicating that arbitrary subspace selection is detrimental for realistic heterogeneous tasks.

ODC achieves *optimal* performance under ideal conditions (**SameSpan-Orth**), shows *sensitivity* to orthonormality violations, yet remains *robust* against subspace misalignment provided orthonormality is maintained. Across diverse tasks, ODC consistently matches or exceeds DC baselines and offers privacy–utility trade-offs competitive with those of DP and FL approaches. Table 12 provides a qualitative summary comparing these representative PPML frameworks. The observations discussed above strongly advocate enforcing orthonormal secret bases (via PCA, SVD, or QR) while tolerating moderate span misalignment to effectively handle the heterogeneity of realistic data.

Table 12: Qualitative comparison of representative PPML frameworks.

Framework	**Communication rounds for training**	**Typical accuracy**[*]	**Privacy guarantee**
DP (additive perturbation)	One user → analyst	Low	Rigorous (ε, δ)–DP
Federated Learning	Iterative bidirectional rounds until convergence	High	No intrinsic formalism[†]
ODC (this work)	One user → analyst+ one user-wide anchor broadcast	Medium[$]	Rigorous under the semi-honest model

[*] Relative rankings derived empirically; specific accuracies vary by task.

[†] FL can incorporate DP externally, often reducing accuracy.

[$] Performance approaches FedAvg under strict orthonormality.

8. Empirics on Concordance

Theoretically, orthogonally concordant change-of-basis matrices are expected to preserve downstream performance in distance-based models. However, in practice, violations of assumptions, finite data, and non-distance-based architectures can introduce performance variability. Here, we empirically evaluate whether orthogonal concordance (ODC) offers a practical advantage over weak concordance (Imakura-DC).

8.1. Experimental Design

The methods evaluated are:

- **Imakura-random**: Change-of-basis matrices constructed via **Algorithm 2**, using a uniformly random matrix for $\boldsymbol{R}$.
- **Imakura-identity**: Change-of-basis matrices constructed via **Algorithm 2**, with $\boldsymbol{R} = \boldsymbol{I}$.
- **ODC-random**: Change-of-basis matrices constructed via **Algorithm 4**, using a uniformly random orthogonal matrix for $\boldsymbol{O}$ (Haar distribution).
- **ODC-identity**: Change-of-basis matrices constructed via **Algorithm 4** with $\boldsymbol{O} = \boldsymbol{I}$.

We follow the canonical DC protocol with $c = 100$ users, each providing $n_i = 100$ samples. The anchor matrix $\boldsymbol{A} \in \mathbb{R}^{1000\times784}$ is generated as a uniformly random matrix.

Experiments are conducted on standard image classification datasets (MNIST [50], Fashion-MNIST [51]). For each dataset, we train (i) a Support Vector Machine (SVM), representative of distance-based classifiers, and (ii) a single-hidden-layer MLP with 256 ReLU neurons. Images are flattened into 28×28 vectors and normalized to the interval $[0, 1]$. The reported performance metrics are the mean testing accuracy along with the 95% confidence interval obtained using SVM and MLP models across 100 independent runs. To assess statistical significance, we perform paired, one-sided t-tests at the 5% level, testing the null hypothesis that the *random* variant performs *no worse* than the identity variant. Results are presented in Tables 13 and 14.

All other experimental settings (e.g., hardware, software, and implementation details) are identical to those described in § 7, unless otherwise stated.

8.2. Results and Discussion

Table 13: Evaluation of the practical significance of orthogonal concordance (**ODC**) compared to weak concordance (**Imakura**) under the **SameSpan-Orth** condition on MNIST and Fashion-MNIST with SVM and MLP classifiers. The table reports mean accuracies (%) ± margin of error (95% confidence interval, 100 runs each) for identity and random matrix choices, together with the accuracy difference between random and identity conditions (random minus identity) and the paired Cohen's d for the same comparison. Negative Δ and d indicate that the random matrix underperforms the identity. Statistical significance is indicated by * for $p < 0.05$, based on one-sided (left-tailed) paired t-tests testing the null hypothesis that $\mu_{\text{rand}} \geq \mu_{\text{iden}}$; $^-$ denotes non-significant differences.

Condition	Dataset	Classifier	Imakura-Identity	Imakura-Random	Imakura-Δ [%]	Imakura-d	ODC-Identity	ODC-Random	ODC-Δ [%]	ODC-d
SameSpan-Orth	MNIST	SVM	96.20 ± 0.00	92.79 ± 0.12	-3.42^*	−5.81	96.00 ± 0.00	96.00 ± 0.00	$+0.00^-$	+0.00
	MNIST	MLP	96.17 ± 0.05	95.34 ± 0.10	-0.82^*	−1.50	96.00 ± 0.00	96.41 ± 0.05	$+0.41^-$	+1.58
	Fashion-MNIST	SVM	86.21 ± 0.01	82.60 ± 0.16	-3.61^*	−4.41	84.30 ± 0.00	84.31 ± 0.00	$+0.00^-$	+0.00
	Fashion-MNIST	MLP	86.62 ± 0.09	85.28 ± 0.16	-1.35^*	−1.50	85.70 ± 0.00	86.47 ± 0.12	$+0.77^-$	+1.28

Table 13 reports the results for the **SameSpan-Orth** setting, in which **Assumption 4.1** and **Assumption 5.1** are simultaneously satisfied, so that both weak and orthogonal concordance hold. In this ideal regime, Imakura's weak-concordance method exhibits a strong dependence on the choice of right factor $\boldsymbol{R}$. Replacing the identity ($\boldsymbol{R} = \boldsymbol{I}$) with a uniformly random invertible matrix consistently reduces accuracy by about 0.8–3.6 percentage points across all four tasks, with large negative effect sizes ($d \leq -1.5$) and statistically significant differences (all entries in the Imakura-Δ column are marked with *). Thus, even when weak concordance is theoretically guaranteed, the particular target basis has a substantial practical impact on downstream performance.

By contrast, ODC is essentially invariant to the choice of orthogonal matrix O under the same conditions. For the SVM classifiers, **ODC-Identity** and **ODC-Random** yield numerically identical accuracies. For the MLPs, the

Table 14: Evaluation of concordance under different span conditions (**SameSpan**, **DiffSpan**, **DiffSpan-Orth**), where the theoretical assumptions are partially violated, on MNIST and Fashion-MNIST using SVM and MLP classifiers. The table reports mean accuracies (%) ± margin of error (95% confidence interval, 100 runs each) for identity and random matrix choices, together with the accuracy difference between random and identity conditions (random minus identity) and the paired Cohen's d for the same comparison. Negative Δ and d indicate that the random matrix underperforms the identity. Statistical significance is indicated by * for $p < 0.05$, based on one-sided (left-tailed) paired t-tests testing the null hypothesis that $\mu_{\text{rand}} \geq \mu_{\text{iden}}$; $^-$ denotes non-significant differences.

Condition	Dataset	Classifier	Imakura-Identity	Imakura-Random	Imakura-Δ [%]	Imakura-d	ODC-Identity	ODC-Random	ODC-Δ [%]	ODC-d
SameSpan	MNIST	SVM	96.20 ± 0.00	92.99 ± 0.11	−3.21*	−5.70	87.00 ± 0.00	87.00 ± 0.00	+0.00$^-$	+0.00
	MNIST	MLP	96.21 ± 0.05	95.40 ± 0.09	−0.81*	−1.49	89.30 ± 0.00	90.83 ± 0.17	+1.53$^-$	+1.81
	Fashion-MNIST	SVM	86.20 ± 0.00	82.77 ± 0.14	−3.43*	−4.80	77.80 ± 0.00	77.80 ± 0.00	+0.00$^-$	+0.00
	Fashion-MNIST	MLP	86.51 ± 0.11	85.24 ± 0.16	−1.26*	−1.22	79.10 ± 0.00	78.45 ± 0.20	−0.65*	−0.63
DiffSpan-Orth	MNIST	SVM	95.40 ± 0.00	91.90 ± 0.13	−3.50*	−5.38	95.40 ± 0.00	95.40 ± 0.00	+0.00$^-$	+0.00
	MNIST	MLP	95.55 ± 0.07	94.69 ± 0.11	−0.86*	−1.33	95.90 ± 0.00	95.52 ± 0.06	−0.38*	−1.20
	Fashion-MNIST	SVM	84.30 ± 0.00	81.36 ± 0.16	−2.94*	−3.63	81.30 ± 0.00	81.30 ± 0.00	+0.00$^-$	+0.00
	Fashion-MNIST	MLP	84.37 ± 0.10	83.45 ± 0.16	−0.92*	−0.98	84.70 ± 0.00	83.47 ± 0.16	−1.23*	−1.53
DiffSpan	MNIST	SVM	94.38 ± 0.01	91.24 ± 0.13	−3.13*	−4.65	83.90 ± 0.00	83.90 ± 0.00	+0.00$^-$	+0.00
	MNIST	MLP	94.65 ± 0.06	93.76 ± 0.13	−0.90*	−1.16	87.50 ± 0.00	87.87 ± 0.22	+0.37$^-$	+0.32
	Fashion-MNIST	SVM	83.20 ± 0.00	80.62 ± 0.15	−2.58*	−3.33	70.50 ± 0.00	70.50 ± 0.00	+0.00$^-$	+0.00
	Fashion-MNIST	MLP	83.80 ± 0.10	82.61 ± 0.19	−1.19*	−1.05	77.80 ± 0.00	72.66 ± 0.37	−5.15*	−2.69

differences remain within 0.8 percentage points and are not flagged as significant by the one-sided t-tests (all entries carry the $^-$ mark). In three out of four cases, the random orthogonal matrix even attains a slightly higher mean accuracy than the identity, although we do not test for the superiority of the random choice. These observations are consistent with **Theorem 5.3**: under **Assumption 5.1**, all parties' representations are aligned up to a common orthogonal transform, implying exact invariance for distance-based models, such as SVMs, and near-invariance in practice, even for MLPs.

Table 14 examines how these conclusions change when the secret-basis assumptions are partially violated. For Imakura's method, the qualitative picture does not change: across all three conditions (**SameSpan**, **DiffSpan**, **DiffSpan-Orth**), **Imakura-Random** underperforms **Imakura-Identity** by roughly 0.8–3.5 percentage points, with consistently large negative Cohen's d and statistically significant differences. This confirms that the instability of weak concordance with respect to the choice of $\boldsymbol{R}$ is intrinsic and persists even when its theoretical assumptions are not met.

For ODC, the behavior depends strongly on which parts of **Assumption** 5.1 are relaxed. When the bases share an identical span but lose orthonormality (**SameSpan**), the absolute accuracy of **ODC-Identity** is substantially degraded compared to the **SameSpan-Orth** case (drops of about 6–7 percentage points across both datasets and classifiers), highlighting orthonormality as the key structural requirement for ODC. In this regime, the additional effect of choosing a random orthogonal matrix $\boldsymbol{O}$ is relatively modest (within about ±1.5 percentage points) and task-dependent: for MNIST, the random choice slightly improves accuracy, whereas for Fashion-MNIST, it leads to small but statistically significant decreases.

When orthonormality is preserved but the spans differ (**DiffSpan-Orth**), ODC remains quite stable. For SVMs, **ODC-Identity** and **ODC-Random** are equivalent, and for MLPs, the degradation of **ODC-Random** relative to **ODC-Identity** is at most about 1.2 percentage points. This indicates that ODC tolerates moderate subspace misalignment as long as orthonormality is enforced. Finally, when both identical-span and orthonormality assumptions are violated simultaneously (**DiffSpan**), ODC experiences the largest loss of absolute accuracy and becomes most sensitive to the choice of $\boldsymbol{O}$, with differences of up to about 5 percentage points (Fashion-MNIST, MLP). In this regime, the alignment matrices obtained from the Orthogonal Procrustes step no longer correspond to a common global rotation, so a random orthogonal target can indeed be suboptimal.

Taken together, Tables 13 and 14 empirically confirm the theoretical claims of orthogonal concordance. When **Assumption** 5.1 holds, ODC is practically invariant to the choice of the orthogonal target basis, in sharp contrast to the weak-concordance baseline. When the assumptions are relaxed, ODC continues to exhibit smaller and more structured sensitivity to the random matrix than Imakura's method, and its degradation is primarily driven by violations of orthonormality rather than by moderate span mismatches.

9. Empirics on Anchor Construction

The shared anchor dataset $\boldsymbol{A} \in \mathbb{R}^{a \times m}$ is the only object that is common across users, and its intermediate representations $\boldsymbol{A}_i = \boldsymbol{A}\boldsymbol{F}_i$ are the sole inputs used by the analyst to construct the *change-of-basis* matrices $\boldsymbol{G}_i$ (**Algorithm** 1).

Table 15: Test accuracy (%) of DC methods as a function of anchor size a on MNIST and Fashion-MNIST using an MLP classifier. We report the mean accuracy ± 95% confidence interval over 100 iterations.

Dataset	Condition	Method	$a = 196$	392	784	1568
MNIST	**SameSpan-Orth**	Imakura-DC [6]	95.74±0.12	95.92±0.14	95.65±0.14	95.59±0.11
		Kawakami-DC [12]	94.78±0.15	94.63±0.21	94.11±0.24	93.41±0.25
		ODC	**96.03±0.13**	**96.04±0.14**	**95.90±0.12**	**95.96±0.12**
	DiffSpan-Orth	Imakura-DC [6]	93.33±0.14	94.49±0.17	94.61±0.14	94.47±0.13
		Kawakami-DC [12]	91.81±0.21	92.27±0.27	91.22±0.34	90.32±0.26
		ODC	**94.02±0.15**	**94.83±0.15**	**94.92±0.14**	**95.15±0.12**
Fashion-MNIST	**SameSpan-Orth**	Imakura-DC [6]	**86.08±0.23**	86.22±0.22	85.91±0.22	85.70±0.19
		Kawakami-DC [12]	84.32±0.29	83.93±0.26	83.23±0.31	82.69±0.24
		ODC	86.05±0.23	**86.23±0.20**	**86.35±0.23**	**86.07±0.18**
	DiffSpan-Orth	Imakura-DC [6]	80.22±0.35	**82.19±0.30**	**82.57±0.29**	82.76±0.25
		Kawakami-DC [12]	78.82±0.37	79.71±0.26	79.79±0.30	80.42±0.25
		ODC	**80.36±0.32**	81.81±0.27	82.24±0.26	**82.88±0.22**

While §6 established that ODC's alignment runtime scales favorably with the anchor size a, a natural question is whether *how we construct* $\boldsymbol{A}$ (e.g., synthetic vs. public anchors, and the choice of a) also affects downstream *utility* and the practical *privacy–efficiency* trade-off.

9.1. Experimental Design

Throughout, we keep the secret bases $\boldsymbol{F}_i$ and downstream models fixed while varying (i) the anchor size a and (ii) the distribution from which the rows of $\boldsymbol{A}$ are sampled. We compare the proposed ODC framework against two existing basis-alignment schemes, Imakura-DC and Kawakami-DC, and report the mean ±95% confidence interval over repeated runs.

For MNIST and Fashion-MNIST we reuse the setup of § 7, draw each row of $\boldsymbol{A}$ i.i.d. from Unif$[0, 1)$, and vary the anchor size over $a \in \{196, 392, 784, 1568\}$. Table 15 reports MLP test accuracy for Imakura-DC, Kawakami-DC, and ODC under the secret-basis conditions **SameSpan-Orth** and **DiffSpan-Orth**.

We next investigate how the *source* of the anchor interacts with a on AMES. Following standard DC practice, we consider two constructions of $\boldsymbol{A}$: (i) a synthetic anchor whose rows are sampled i.i.d. from Unif$[0, 1)$ ("Uniform"), and (ii) a domain-matched anchor [17] obtained by sampling unlabeled molecular fingerprints from PubChem [52]. For each construction, we vary $a \in \{512, 1024, 2048, 4096\}$ and evaluate ODC, Imakura-DC, and Kawakami-DC under **SameSpan-Orth** and **DiffSpan-Orth**. Table 16 reports ROC-AUC and PR-AUC.

All other experimental settings (e.g., hardware, software, and implementation details) are identical to those described in § 7, unless otherwise stated.

9.2. Results and Discussion

Image classification (MNIST, Fashion-MNIST). Under **SameSpan-Orth**, all three DC methods are remarkably stable in a. For ODC, the accuracy ranges from 95.90% to 96.04% on MNIST and from 86.05% to 86.35% on Fashion-MNIST as a increases by a factor of eight. Imakura-DC closely tracks ODC (within about 0.3 percentage points on average), while Kawakami-DC is consistently 1–3 points worse. Thus, once a moderately exceeds the latent dimension ℓ, further enlarging $\boldsymbol{A}$ has a negligible impact on image-level utility for any of the DC schemes.

Under **DiffSpan-Orth**, accuracies drop for all methods due to span mismatch between parties, but the dependence on a remains weak. Increasing a from 196 to 1568 improves ODC by about 1.1 percentage points on MNIST (94.02% → 95.15%) and 2.5 points on Fashion-MNIST (80.36% → 82.88%), with Imakura-DC exhibiting similar gains (e.g., 93.33% → 94.47% on MNIST, 80.22% → 82.76% on Fashion-MNIST). Kawakami-DC lags further behind—roughly 2–4 points below ODC across all a in this harder regime. Averaged over anchor sizes, ODC and Imakura-DC are essentially tied (differences ≤ 0.5 points), and both substantially outperform Kawakami-DC. For

Table 16: ODC and baseline DC performance on AMES as a function of anchor size a and anchor source. We report mean ROC-AUC / PR-AUC (%) ± 95% confidence intervals over 100 iterations. "Uniform" uses synthetic anchors with rows sampled i.i.d. from Unif$[0, 1)$; "PubChem" uses unlabeled compounds from PubChem as rows of $\boldsymbol{A}$.

Condition	Anchor	Method	a = 512	1024	2048	4096
			ROC-AUC (%) ↑			
SameSpan-Orth	Uniform	Imakura-DC [6]	88.23±0.13	87.90±0.18	86.93±0.33	84.66±0.52
		Kawakami-DC [12]	87.38±0.17	85.89±0.40	83.07±0.42	81.63±0.24
		ODC	**88.64±0.10**	**88.59±0.12**	**88.68±0.11**	**88.71±0.10**
	PubChem	Imakura-DC [6]	88.10±0.10	88.09±0.14	87.89±0.14	87.28±0.21
		Kawakami-DC [12]	86.91±0.19	86.87±0.19	85.88±0.33	84.06±0.42
		ODC	**88.73±0.09**	**88.63±0.12**	**88.61±0.13**	**88.66±0.10**
DiffSpan-Orth	Uniform	Imakura-DC [6]	59.11±0.28	60.81±0.30	62.22±0.34	63.29±0.37
		Kawakami-DC [12]	59.11±0.29	61.39±0.34	64.10±0.37	65.98±0.42
		ODC	**61.84±0.30**	**64.34±0.31**	**66.78±0.28**	**68.65±0.30**
	PubChem	Imakura-DC [6]	77.57±0.19	80.60±0.17	81.94±0.16	82.31±0.15
		Kawakami-DC [12]	74.36±0.21	77.30±0.22	78.28±0.22	78.65±0.21
		ODC	**84.51±0.13**	**85.68±0.11**	**86.37±0.11**	**86.60±0.11**
			PR-AUC (%) ↑			
SameSpan-Orth	Uniform	Imakura-DC [6]	89.67±0.13	89.36±0.17	88.56±0.34	86.31±0.54
		Kawakami-DC [12]	88.97±0.17	87.55±0.42	84.69±0.44	83.02±0.25
		ODC	**89.96±0.10**	**89.91±0.12**	**90.00±0.11**	**90.00±0.10**
	PubChem	Imakura-DC [6]	89.61±0.10	89.55±0.13	89.41±0.13	88.87±0.21
		Kawakami-DC [12]	88.52±0.18	88.48±0.19	87.47±0.36	85.49±0.47
		ODC	**90.08±0.10**	**89.85±0.11**	**89.89±0.12**	**89.95±0.09**
DiffSpan-Orth	Uniform	Imakura-DC [6]	63.05±0.29	64.87±0.30	66.11±0.31	67.11±0.34
		Kawakami-DC [12]	63.16±0.29	65.36±0.31	67.78±0.34	69.74±0.40
		ODC	**65.51±0.31**	**67.87±0.31**	**70.33±0.28**	**72.19±0.29**
	PubChem	Imakura-DC [6]	80.56±0.19	83.16±0.17	84.23±0.16	84.55±0.14
		Kawakami-DC [12]	77.71±0.21	80.27±0.20	80.98±0.19	81.24±0.19
		ODC	**86.51±0.13**	**87.53±0.12**	**88.15±0.11**	**88.32±0.11**

image classification, anchor size is therefore a second-order effect compared to the choice of basis-alignment method and the validity of the span assumptions.

Molecular classification (AMES) and anchor source. Under **SameSpan-Orth**, all three methods lie close to the centralized oracle, and both the size and source of $\boldsymbol{A}$ have only a mild influence. ODC remains the strongest method, with ROC-AUC around 88.6% and PR-AUC around 90.0% across all anchor sizes and both anchor sources, while Imakura-DC trails by roughly 1–2 points and Kawakami-DC by 3–4 points on average. The variation of ODC's scores with a is extremely small (at most 0.1 points in ROC-AUC), confirming that when the span assumptions are satisfied, anchor construction is largely irrelevant for utility.

Under **DiffSpan-Orth**, anchor design becomes critical. With a synthetic Uniform anchor, all methods degrade substantially relative to the centralized model: for ODC, ROC-AUC ranges from 61.84% to 68.65% and PR-AUC from 65.51% to 72.19% as a grows from 512 to 4096, while Imakura-DC and Kawakami-DC are roughly 2–5 points below ODC at each anchor size. Switching from Uniform to a PubChem anchor dramatically stabilizes all three methods, but especially ODC. For instance, at $a = 4096$ ODC attains 86.60% ROC-AUC and 88.32% PR-AUC, compared with 82.31% / 84.55% for Imakura-DC and 78.65% / 81.24% for Kawakami-DC. Across anchor sizes, PubChem anchors

improve ODC by roughly 18–22 points over its Uniform counterpart in both ROC-AUC and PR-AUC, while yielding gains of 15–20 points for the baselines. In this challenging span-mismatched regime, the proposed ODC framework consistently achieves the highest scores, outperforming Imakura-DC by approximately 4–7 points and Kawakami-DC by about 8–10 points.

Statistical analysis. We now quantify how strongly anchor construction affects performance using formal hypothesis tests. For the *anchor size* experiments, we test, for each task, span condition, method, and metric, the null hypothesis

$$H_{0,\text{size}}: \quad \mathbb{E}[\text{metric} \mid a = a_1] = \cdots = \mathbb{E}[\text{metric} \mid a = a_K], \tag{93}$$

i.e., that the mean performance is identical across all anchor sizes $a \in \{a_1, \ldots, a_K\}$. For the AMES anchor *source* experiments (Uniform vs. PubChem), we fit two-way ANOVA models with factors anchor source and anchor size and test, for each span condition, method, and metric, the null

$$H_{0,\text{source}}: \quad \mathbb{E}[\text{metric} \mid \text{Uniform}] = \mathbb{E}[\text{metric} \mid \text{PubChem}], \tag{94}$$

conditional on a. In all cases we report the F-statistic, p-value, and Cohen's f ($f \approx 0.10$ small, $f \approx 0.25$ medium, $f \approx 0.40$ large).

Tables 17 and 18 summarize the one-way ANOVA results for anchor size. On MNIST and Fashion-MNIST under **SameSpan-Orth**, ODC shows no statistically significant dependence on a ($p > 0.15$, $f < 0.12$ on both datasets), while Imakura-DC exhibits small effects ($p < 0.005$, $f \approx 0.18$–0.19) and Kawakami-DC moderate ones ($p < 10^{-16}$, $f \approx 0.45$–0.47). Under **DiffSpan-Orth**, the null $H_{0,\text{size}}$ is rejected for all three methods, with large effect sizes (e.g., for ODC, $f \approx 0.57$ on MNIST and 0.67 on Fashion-MNIST), although the corresponding changes in mean accuracy remain modest (on the order of 1–3 percentage points as a increases; Table 15). Thus, in the well-specified image regime, ODC is essentially invariant to a, whereas the contemporary methods show mild-to-moderate sensitivity; in the span-mismatched regime, all DC methods benefit from larger anchors, with ODC typically achieving the best accuracies.

On AMES, the one-way ANOVA reveals a similar pattern (Table 18). Under **SameSpan-Orth** with synthetic Uniform anchors, ODC again appears insensitive to a ($p = 0.40$ and 0.66 for ROC-AUC and PR-AUC, $f \leq 0.09$), while Imakura-DC and Kawakami-DC show strong statistical dependence on a with large effect sizes (e.g., for Uniform ROC-AUC, $f \approx 0.83$ for Imakura-DC and 1.37 for Kawakami-DC). The same trend holds for PubChem anchors: ODC has at most a small effect of a (e.g., PR-AUC $p = 0.017$, $f = 0.16$), whereas Imakura-DC and Kawakami-DC exhibit medium-to-large effects ($f \approx 0.39$–0.76). Under **DiffSpan-Orth**, $H_{0,\text{size}}$ is overwhelmingly rejected for all three methods and both anchor sources, with very large Cohen's f values ($f \gtrsim 0.95$ and often > 1.5), reflecting the substantial improvements in ROC-AUC and PR-AUC as a grows from 512 to 4096 (Table 16). Overall, anchor size is practically negligible for ODC in the well-specified setting, but it can matter substantially for Imakura-DC and Kawakami-DC and for *all* methods under span mismatch.

To isolate the effect of anchor *source* on AMES, Table 19 reports two-way ANOVA results for the main effect of anchor source (Uniform vs. PubChem), controlling for a. Under **SameSpan-Orth**, ODC does not distinguish between Uniform and PubChem anchors (ROC-AUC: $p = 0.87$, $f = 0.01$; PR-AUC: $p = 0.53$, $f = 0.02$), whereas Imakura-DC and Kawakami-DC show statistically significant but small-to-moderate effects ($f \approx 0.33$–0.46), corresponding to absolute differences of roughly 1–1.5 percentage points between the two anchor distributions. Under **DiffSpan-Orth**, the anchor source has an extremely large impact for all three methods: for ROC-AUC and PR-AUC the null $H_{0,\text{source}}$ is rejected with F-statistics on the order of 10^4 and Cohen's f between 4.8 and 8.9, consistent with the $\approx$ 13–20 point gains observed when replacing a synthetic Uniform anchor with a domain-matched PubChem anchor. Among the three DC schemes, ODC consistently achieves the largest absolute improvements with PubChem (around 18–20 points in ROC-AUC and PR-AUC), and is also the most robust to anchor size when the span assumptions are satisfied.

Privacy considerations. Across all experiments, anchor construction does not alter the formal privacy guarantees of DC. The anchor dataset $\boldsymbol{A}$ is either synthetic or drawn from public unlabeled data, and only its projections $\boldsymbol{A}\boldsymbol{F}_i$ and the user-side representations $\tilde{\boldsymbol{X}}_i = \boldsymbol{X}_i\boldsymbol{F}_i$ participate in the collaboration protocol. As a result, the threat surfaces analyzed in § 2.2 (e.g., collusion and reconstruction attacks) remain governed by the secret bases $\boldsymbol{F}_i$ and the latent dimension ℓ, rather than by the particular sampling distribution or size of $\boldsymbol{A}$. The anchor ablations in this section, therefore, show that anchor design is a powerful knob for improving utility in difficult span-mismatched settings—especially for ODC—without introducing additional privacy risks.

Table 17: One-way ANOVA on the effect of anchor size a on MNIST and Fashion-MNIST performance with Uniform anchors. For each span condition and method, we report the F-statistic, p-value, and Cohen's f for $H_{0,\text{size}}$.

Dataset	Condition	Method	Effect of anchor size a		
			F	p	Cohen's f
MNIST	**SameSpan-Orth**	Imakura-DC [6]	5.04	0.002	0.19
		Kawakami-DC [12]	32.37	6.79×10^{-19}	0.47
		ODC	1.05	0.371	0.09
	DiffSpan-Orth	Imakura-DC [6]	59.41	3.38×10^{-32}	0.64
		Kawakami-DC [12]	39.70	1.10×10^{-22}	0.52
		ODC	47.20	2.05×10^{-26}	0.57
Fashion-MNIST	**SameSpan-Orth**	Imakura-DC [6]	4.44	0.004	0.18
		Kawakami-DC [12]	28.10	1.41×10^{-16}	0.45
		ODC	1.74	0.158	0.11
	DiffSpan-Orth	Imakura-DC [6]	59.30	5.56×10^{-32}	0.65
		Kawakami-DC [12]	19.76	5.33×10^{-12}	0.38
		ODC	63.44	7.54×10^{-34}	0.67

10. Conclusion

In this paper, we revisited the Data Collaboration (DC) paradigm and identified a key gap between existing theory and practice. Although weak concordance guarantees that parties can be aligned to a common subspace, downstream performance can still vary substantially with the analyst's *choice of target basis*. To resolve this instability, we proposed **Orthonormal Data Collaboration (ODC)**, which enforces orthonormality on both secret and target bases. Under this constraint, basis alignment reduces exactly to the classical Orthogonal Procrustes Problem [14], yielding a closed-form solution and improving the alignment complexity from

$$O(\min\{a(c\ell)^2,\ a^2c\ell\}) \ \rightarrow \ O(ac\ell^2), \tag{95}$$

where a is the anchor size, ℓ the latent dimension, and c the number of parties.

Our empirical results validate these computational gains. On the controlled efficiency benchmark (Fig. 6), ODC consistently outperformed contemporary DC alignments with speed-ups ranging from roughly 6× to over 100×; for example, at $a = 20{,}000$ the median runtime dropped from ≈ 50 s (baselines) to 0.47 s (ODC), i.e., more than two orders of magnitude. Moreover, the scaling experiments demonstrate that ODC remains practical even as the number of users increases (Fig. 6c), rendering the alignment phase typically negligible in realistic deployments.

Beyond speed, ODC's central benefit is *stability*. We proved that ODC's orthogonal change-of-basis matrices satisfy *orthogonal concordance*, aligning all parties' representations up to a common orthogonal transform and thereby preserving distances and inner products. This theoretical invariance is reflected in the concordance experiments (Tables 13–14): replacing the identity target with a random *invertible* target in Imakura-DC causes consistent and statistically significant accuracy drops (up to 3–4 percentage points), whereas ODC is essentially invariant to the choice of orthogonal target—exactly so for SVMs and nearly so for MLPs under the orthonormal setting.

Across application benchmarks, ODC matches or improves upon existing DC methods when orthonormality holds (Tables 7–9). Under the realistic **DiffSpan-Orth** condition (heterogeneous spans but orthonormal bases), ODC remains competitive and often exceeds the DC baselines on biomedical tasks, while the ablations also make the practical requirement clear: violating orthonormality leads to marked degradation for ODC (Tables 7–9). This directly motivates the use of standard orthonormal basis construction (e.g., PCA/SVD/QR), which is already common in DC pipelines.

Finally, ODC compares favorably to representative PPML baselines while retaining DC's one-shot communication pattern. On CelebA, DC-style projections provide strong visual obfuscation and reduce FaceNet-based identifiability toward chance (Table 10), while maintaining accuracy competitive with high-ε DP noise and substantially better

Table 18: One-way ANOVA on the effect of anchor size a on AMES performance with Uniform and Pubchem anchors. For each span condition, metric, and method we report the F-statistic, p-value, and Cohen's f for $H_{0,\text{size}}$.

Condition	Anchor	Method	Effect of anchor size a		
			F	p	Cohen's f
ROC-AUC (%)					
SameSpan-Orth	Uniform	Imakura-DC [6]	91.55	5.08×10^{-45}	0.83
		Kawakami-DC [12]	248.96	9.34×10^{-91}	1.37
		ODC	0.98	0.404	0.09
	PubChem	Imakura-DC [6]	25.54	4.00×10^{-15}	0.44
		Kawakami-DC [12]	77.09	2.73×10^{-39}	0.76
		ODC	0.93	0.425	0.08
DiffSpan-Orth	Uniform	Imakura-DC [6]	119.68	3.54×10^{-55}	0.95
		Kawakami-DC [12]	278.68	3.30×10^{-97}	1.45
		ODC	385.15	5.19×10^{-117}	1.71
	PubChem	Imakura-DC [6]	626.14	6.98×10^{-150}	2.18
		Kawakami-DC [12]	318.73	3.34×10^{-105}	1.55
		ODC	249.27	7.94×10^{-91}	1.37
PR-AUC (%)					
SameSpan-Orth	Uniform	Imakura-DC [6]	77.97	1.19×10^{-39}	0.77
		Kawakami-DC [12]	243.30	1.80×10^{-89}	1.36
		ODC	0.54	0.656	0.06
	PubChem	Imakura-DC [6]	20.33	2.83×10^{-12}	0.39
		Kawakami-DC [12]	74.20	4.24×10^{-38}	0.75
		ODC	3.42	0.017	0.16
DiffSpan-Orth	Uniform	Imakura-DC [6]	121.81	6.69×10^{-56}	0.96
		Kawakami-DC [12]	273.49	4.08×10^{-96}	1.44
		ODC	364.78	1.47×10^{-113}	1.66
	PubChem	Imakura-DC [6]	465.59	1.96×10^{-129}	1.88
		Kawakami-DC [12]	255.57	3.11×10^{-92}	1.39
		ODC	191.34	1.12×10^{-76}	1.20

than stricter DP settings (Table 9). On eICU regression, ODC under **DiffSpan-Orth** achieves RMSE close to FedAvg while avoiding iterative communication (Table 11). Section 9 further shows that ODC's behavior with respect to anchor design is well-behaved: when the span assumptions hold, it is essentially insensitive to anchor size and source—image accuracies move by at most about 0.3 percentage points across an 8× sweep in a (Table 15), and AMES ROC-/PR-AUC under **SameSpan-Orth** are nearly flat in both a and between synthetic and PubChem anchors (Table 16)—whereas under the more realistic **DiffSpan-Orth** regime, anchor construction becomes a powerful lever, with domain-matched PubChem anchors improving ODC by roughly 18–22 ROC-/PR-AUC points over synthetic anchors on AMES (Table 16) without changing DC's semi-honest privacy model.

Promising future directions include strengthening privacy beyond the semi-honest model (e.g., collusion-robust variants and principled DP integration on released representations), extending orthonormal alignment ideas to nonlinear mappings and partially overlapping feature spaces, and *deepening* deployment guidelines (e.g., automatic strategies for selecting ℓ and anchor schemes under task-specific constraints on utility, communication, and attack surfaces).

Table 19: Two-way ANOVA on the effect of anchor source (Uniform vs. PubChem) on AMES performance, controlling for anchor size a. For each span condition, method, and metric, we report the main-effect F-statistic, p-value, and Cohen's f for the null hypothesis $H_{0,\text{source}}$ that Uniform and PubChem anchors yield identical mean performance.

Condition	Metric	Method	Effect of anchor source		
			F	p	Cohen's f
SameSpan-Orth	PR-AUC ↑	Imakura-DC [6]	88.18	6.20×10^{-20}	0.33
		Kawakami-DC [12]	143.53	1.63×10^{-30}	0.43
		ODC	0.40	0.526	0.02
	ROC-AUC ↑	Imakura-DC [6]	97.21	1.04×10^{-21}	0.35
		Kawakami-DC [12]	163.15	4.18×10^{-34}	0.45
		ODC	0.03	0.868	0.01
DiffSpan-Orth	PR-AUC ↑	Imakura-DC [6]	39,598.38	$< 10^{-300}$	7.07
		Kawakami-DC [12]	18,249.18	$< 10^{-300}$	4.80
		ODC	52,065.06	$< 10^{-300}$	8.11
	ROC-AUC ↑	Imakura-DC [6]	42,639.30	$< 10^{-300}$	7.34
		Kawakami-DC [12]	18,904.45	$< 10^{-300}$	4.89
		ODC	63,204.41	$< 10^{-300}$	8.93

10.1. Practical deployment and systems considerations

While ODC is motivated by a mathematical gap in basis alignment, its design targets *deployable* cross-silo workflows where parties cannot centralize raw records and where iterative communication (e.g., FL-style rounds) is expensive or operationally infeasible. A semi-realistic end-to-end deployment pipeline typically consists of three stages: (i) a one-off anchor bootstrapping step in which the consortium agrees on a shared anchor $\boldsymbol{A}$ and each site constructs its secret basis $\boldsymbol{F}_i$; (ii) a single DC round in which users compute and upload $(\tilde{\boldsymbol{X}}_i, \boldsymbol{A}_i, \boldsymbol{L}_i)$, and the analyst performs ODC alignment and model training; and (iii) a deployment step where each site receives $(\boldsymbol{G}_i, h)$ and applies $h(\boldsymbol{Y}_i\boldsymbol{F}_i\boldsymbol{G}_i)$ to local test data. Table 20 distills the main engineering and governance decisions that a practitioner must finalize. Below, we highlight practical considerations that complement our analyses of complexity, memory, and communication (e.g., § 2.3).

Cold-start anchor generation (one-off). A deployment must decide how to instantiate the shared anchor matrix $\boldsymbol{A} \in \mathbb{R}^{a\times m}$ at "cold start," corresponding to the initial stage above. Common options are: *(i) synthetic anchors* (e.g., i.i.d. random or dummy feature vectors), *(ii) public/unlabeled anchors* from an external source (domain-dependent), or *(iii) seed-based anchors* where all parties deterministically generate the same $\boldsymbol{A}$ from a shared PRNG seed (avoiding repeated cross-silo distribution). Independently of the source, the engineering requirement is to satisfy the rank and conditioning assumptions used by DC/ODC (e.g., $\operatorname{rank}(\boldsymbol{A}) = m$ and typically $a > m$), while keeping $\boldsymbol{A}$ non-sensitive (not derived from private records) to avoid governance complications. These choices and their documentation are captured in the "Anchor plan" row of Table 20.

Resource constraints: compute, memory, and network. ODC's alignment workload is dominated by per-party $\ell \times \ell$ Procrustes/SVD computations and $\ell \times a$ by $a \times \ell$ products, which are parallelizable and friendly to modern BLAS kernels. Peak memory at the analyst is driven by storing the transformed anchors $\{\boldsymbol{A}_i\}_{i=1}^{c}$ (order $\Theta(ac\ell)$), as indicated in the "Compute/memory" row of Table 20. For networking, DC/ODC remains a one-shot protocol: parties upload $(\tilde{\boldsymbol{X}}_i, \boldsymbol{A}_i, \boldsymbol{L}_i)$ once and receive $(\boldsymbol{G}_i, h)$ once. The practical traffic budget can be estimated directly from the communication model in § 2.3; the existing healthcare example (100 hospitals with a ResNet-50) illustrates how this translates to concrete GB-scale volumes and highlights when DC/ODC is advantageous under bandwidth and RTT constraints. The "Network budget" and "Numerics" rows in Table 20 summarize the associated design choices (precision, quantization, batching, compression).

Cross-sector deployment sketches. The same pipeline applies across sectors: only the local site semantics (data type, governance regime) change. Typical instantiations include:

Table 20: Deployment checklist for ODC/DC (engineering and governance).

Item	What to decide / verify in practice
Threat model	Confirm semi-honest assumptions; decide whether collusion is in scope and whether additional protections are needed (e.g., DP on released representations, encryption in transit/at rest) are required.
Anchor plan	Synthetic/public/seed-based anchor; document cold-start distribution (topology, replication factor, and whether seeding avoids cross-silo transfer).
Basis selection	Enforce orthonormal $\boldsymbol{F}_i$ (PCA/SVD/QR); choose latent dimension ℓ to balance utility against communication volume and collusion surface.
Numerics	Choose precision/quantization (e.g., FP32/FP16/int8) for uplink; validate alignment stability for $\ell \times \ell$ matrices and heterogeneous sites.
Network budget	Estimate uplink/downlink traffic using the DC accounting in § 2.3; plan batching, compression, and tolerance to intermittent or low-bandwidth links.
Compute/memory	Analyst memory is dominated by storing $\boldsymbol{A}_i$ for all $i \in [c]$ (order $\Theta(ac\ell)$); consider streaming or parallelization across sites; document expected wall-clock runtime for alignment and training.

- **Cross-institution consortia (general):** organizations keep data on-prem and share only projected representations for joint modeling; ODC's closed-form alignment supports predictable runtime and avoids multi-round coordination overhead.
- **Medical / multi-hospital:** typical targets include imaging, ICU risk, or outcomes prediction across hospitals with strict data governance. One-shot transfer and explicit communication accounting (§ 2.3) support feasibility checks under hospital network constraints.
- **Power / utilities:** operators can collaborate on forecasting or anomaly detection without exchanging raw operational traces; one-shot protocols reduce persistent connectivity requirements and simplify segmented network operations.
- **Finance:** banks and insurers can collaborate on fraud or risk models while maintaining data locality; deployment often requires strict auditability and controls on what representations and metadata leave each institution.

Compliance and governance considerations (non-exhaustive). ODC inherits the same privacy model as the underlying DC protocol (semi-honest analyst/users), so a deployment should explicitly document (i) the assumed threat model (including whether collusion is in scope), (ii) whether additional protections are required (e.g., encryption in transit/at rest, access control, audit logging, retention limits), and (iii) whether a formal privacy mechanism (e.g., DP layered on released representations) is necessary for the application's regulatory posture. Table 20 can serve as a concise checklist to ensure that these governance and engineering decisions are explicitly documented before a system goes into production.

CRediT authorship contribution statement

Keiyu Nosaka: Conceptualization, Methodology, Software, Validation, Investigation, Data curation, Writing – original draft, Visualization, Funding acquisition, Project administration.
Yamato Suetake: Software, Writing – original draft.
Yuichi Takano: Writing – review & editing, Supervision.
Akiko Yoshise: Writing – review & editing, Supervision, Funding acquisition.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have influenced the work reported in this paper.

Declaration of Generative AI and AI-Assisted Technologies in the Writing Process

During the preparation of this work, the authors employed ChatGPT and Grammarly to enhance the readability and language of the manuscript. Following the use of these services, the authors thoroughly reviewed and edited the content, assuming full responsibility for the final published article.

Data Availability

All datasets used in our experiments are publicly available (see the corresponding citations). The code used to conduct the experiments is available at the following URL:
`https://drive.google.com/drive/folders/1PwXVarZ7mCoDZVzOdYDuZ7jSpILBsXBj?usp=sharing`

Funding Sources

This work was partially supported by the Japan Society for the Promotion of Science (JSPS) KAKENHI under Grant Numbers JP22K18866, JP23K26327, and JP25KJ0701, as well as the Japan Science and Technology Agency (JST) under Grant Number JPMJSP2124.

References

[1] P. Rosati, P. Deeney, M. Cummins, L. van der Werff, T. Lynn, Social media and stock price reaction to data breach announcements: Evidence from us listed companies, Research in International Business and Finance 47 (2019) 458–469.

[2] B. McMahan, E. Moore, D. Ramage, S. Hampson, B. A. y Arcas, Communication-efficient learning of deep networks from decentralized data, in: Artificial Intelligence and Statistics, PMLR, 2017, pp. 1273–1282.

[3] C. Dwork, Differential privacy: A survey of results, in: International Conference on Theory and Applications of Models of Computation, Springer, 2008, pp. 1–19.

[4] K. Wei, J. Li, M. Ding, C. Ma, H. H. Yang, F. Farokhi, S. Jin, T. Q. Quek, H. V. Poor, Federated learning with differential privacy: Algorithms and performance analysis, IEEE transactions on information forensics and security 15 (2020) 3454–3469.

[5] R. Xu, N. Baracaldo, J. Joshi, Privacy-preserving machine learning: Methods, challenges and directions, arXiv preprint arXiv:2108.04417 (2021).

[6] A. Imakura, T. Sakurai, Data collaboration analysis framework using centralization of individual intermediate representations for distributed data sets, ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering 6 (2) (2020) 04020018.

[7] A. Imakura, X. Ye, T. Sakurai, Collaborative data analysis: Non-model sharing-type machine learning for distributed data, in: Knowledge Management and Acquisition for Intelligent Systems: 17th Pacific Rim Knowledge Acquisition Workshop, PKAW 2020, Yokohama, Japan, January 7–8, 2021, Proceedings 17, Springer, 2021, pp. 14–29.

[8] A. Imakura, A. Bogdanova, T. Yamazoe, K. Omote, T. Sakurai, Accuracy and privacy evaluations of collaborative data analysis, Proceedings of the AAAI Conference on Artificial Intelligence (2021).

[9] A. Imakura, T. Sakurai, Y. Okada, T. Fujii, T. Sakamoto, H. Abe, Non-readily identifiable data collaboration analysis for multiple datasets including personal information, Information Fusion 98 (2023) 101826.

[10] H. Yamashiro, K. Omote, A. Imakura, T. Sakurai, Toward the application of differential privacy to data collaboration, IEEE Access PP (2024) 1–1. doi:10.1109/ACCESS.2024.3396146.

[11] A. Imakura, T. Sakurai, Feddcl: a federated data collaboration learning as a hybrid-type privacy-preserving framework based on federated learning and data collaboration, arXiv preprint arXiv:2409.18356 (2024).

[12] Y. Kawakami, Y. Takano, A. Imakura, New solutions based on the generalized eigenvalue problem for the data collaboration analysis, arXiv preprint arXiv:2404.14164 (2024).

[13] K. Nosaka, A. Yoshise, Creating collaborative data representations using matrix manifold optimal computation and automated hyperparameter tuning, in: 2023 IEEE 3rd International Conference on Electronic Communications, Internet of Things and Big Data (ICEIB), IEEE, 2023, pp. 180–185.

[14] P. H. Schönemann, A generalized solution of the orthogonal procrustes problem, Psychometrika 31 (1) (1966) 1–10.

[15] R. Penrose, A generalized inverse for matrices, Proceedings of the Cambridge Philosophical Society 51 (1955) 406–413.

[16] A. Mizoguchi, A. Imakura, T. Sakurai, Application of data collaboration analysis to distributed data with misaligned features, Informatics in Medicine Unlocked 32 (2022) 101013.

[17] A. Mizoguchi, A. Bogdanova, A. Imakura, T. Sakurai, Data collaboration analysis applied to compound datasets and the introduction of projection data to non-iid settings (2023).

[18] T. Nakayama, Y. Kawamata, A. Toyoda, A. Imakura, R. Kagawa, M. Sanuki, R. Tsunoda, K. Yamagata, T. Sakurai, Y. Okada, Data collaboration for causal inference from limited medical testing and medication data (2025). arXiv:2501.06511.
URL https://arxiv.org/abs/2501.06511

[19] Y. Kawamata, R. Motai, Y. Okada, A. Imakura, T. Sakurai, Collaborative causal inference on distributed data, Expert Systems with Applications 244 (2024) 123024. doi:https://doi.org/10.1016/j.eswa.2023.123024.
URL https://www.sciencedirect.com/science/article/pii/S0957417423035261

[20] A. Bogdanova, A. Imakura, T. Sakurai, Dc-shap method for consistent explainability in privacy-preserving distributed machine learning, Human-Centric Intelligent Systems 3 (3) (2023) 197–210.

[21] A. Imakura, R. Tsunoda, R. Kagawa, K. Yamagata, T. Sakurai, Dc-cox: Data collaboration cox proportional hazards model for privacy-preserving survival analysis on multiple parties, Journal of Biomedical Informatics 137 (2023) 104264.

[22] A. Imakura, H. Inaba, Y. Okada, T. Sakurai, Interpretable collaborative data analysis on distributed data, Expert Systems with Applications 177 (2021) 114891. doi:https://doi.org/10.1016/j.eswa.2021.114891.
URL https://www.sciencedirect.com/science/article/pii/S0957417421003328

[23] T. Yanagi, S. Ikeda, N. Sukegawa, Y. Takano, Privacy-preserving recommender system using the data collaboration analysis for distributed datasets, arXiv preprint arXiv:2406.01603 (2024).

[24] Z. Liu, P. Luo, X. Wang, X. Tang, Deep learning face attributes in the wild, in: Proceedings of International Conference on Computer Vision (ICCV), 2015.

[25] H. Nguyen, D. Zhuang, P.-Y. Wu, M. Chang, Autogan-based dimension reduction for privacy preservation, Neurocomputing 384 (2020) 94–103.

[26] J. V. Haxby, J. S. Guntupalli, A. C. Connolly, Y. O. Halchenko, B. R. Conroy, M. I. Gobbini, M. Hanke, P. J. Ramadge, A common, high-dimensional model of the representational space in human ventral temporal cortex, Neuron 72 (2) (2011) 404–416. doi:10.1016/j.neuron.2011.08.026.

[27] A. Lorbert, P. J. Ramadge, Kernel hyperalignment, in: Advances in Neural Information Processing Systems 25, 2012, pp. 1799–1807.

[28] S. Ling, Near-optimal bounds for generalized orthogonal procrustes problem via generalized power method, Applied and Computational Harmonic Analysis 66 (2023) 62–100.

[29] F. Nie, L. Tian, X. Li, Multiview clustering via adaptively weighted procrustes, in: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2018, pp. 2022–2030. doi:10.1145/3219819.3220049.

[30] X. Dong, D. Wu, F. Nie, R. Wang, X. Li, Multi-view clustering with adaptive procrustes on grassmann manifold, Information Sciences 609 (2022) 855–875. doi:10.1016/j.ins.2022.07.089.

[31] C. Wang, S. Mahadevan, Manifold alignment using procrustes analysis, in: Proceedings of the 25th International Conference on Machine Learning, 2008, pp. 1120–1127.

[32] E. Grave, A. Joulin, Q. Berthet, Unsupervised alignment of embeddings with wasserstein procrustes, in: Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics, Vol. 89 of Proceedings of Machine Learning Research, 2019, pp. 1880–1890.

[33] X. Peng, G. Chen, C. Lin, M. Stevenson, Highly efficient knowledge graph embedding learning with orthogonal procrustes analysis, in: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2021, pp. 2364–2375.

[34] R. Iakymchuk, D. Defour, C. Collange, S. Graillat, Reproducible and accurate matrix multiplication, in: International Symposium on Scientific Computing, Computer Arithmetic, and Validated Numerics, Springer, 2015, pp. 126–137.

[35] P.-G. Martinsson, G. Quintana OrtI, N. Heavner, R. Van De Geijn, Householder qr factorization with randomization for column pivoting (hqrrp), SIAM Journal on Scientific Computing 39 (2) (2017) C96–C115.

[36] L. Wang, G. Libert, P. Manneback, Kalman filter algorithm based on singular value decomposition, in: [1992] Proceedings of the 31st IEEE Conference on Decision and Control, IEEE, 1992, pp. 1224–1229.

[37] R. Mahfoudhi, A fast triangular matrix inversion, in: Proceedings of the World Congress on Engineering, Vol. 1, 2012.

[38] K. Chen, L. Liu, Geometric data perturbation for privacy preserving outsourced data mining, Knowledge and information systems 29 (3) (2011) 657–695.

[39] K. Huang, T. Fu, W. Gao, Y. Zhao, Y. Roohani, J. Leskovec, C. W. Coley, C. Xiao, J. Sun, M. Zitnik, Therapeutics data commons: Machine learning datasets and tasks for drug discovery and development, arXiv preprint arXiv:2102.09548 (2021).

[40] B. Becker, R. Kohavi, Adult, UCI Machine Learning Repository, DOI: https://doi.org/10.24432/C5XW20 (1996).

[41] T. J. Pollard, A. E. Johnson, J. D. Raffa, L. A. Celi, R. G. Mark, O. Badawi, The eicu collaborative research database, a freely available multi-center database for critical care research, Scientific data 5 (1) (2018) 1–13.

[42] B. Balle, Y.-X. Wang, Improving the gaussian mechanism for differential privacy: Analytical calibration and optimal denoising, in: International Conference on Machine Learning, PMLR, 2018, pp. 394–403.

[43] C. Xu, F. Cheng, L. Chen, Z. Du, W. Li, G. Liu, P. W. Lee, Y. Tang, In silico prediction of chemical ames mutagenicity, Journal of chemical information and modeling 52 (11) (2012) 2840–2847.

[44] A. Mayr, G. Klambauer, T. Unterthiner, S. Hochreiter, Deeptox: toxicity prediction using deep learning, Frontiers in Environmental Science 3 (2016) 80.

[45] Z. Wu, B. Ramsundar, E. N. Feinberg, J. Gomes, C. Geniesse, A. S. Pappu, K. Leswing, V. Pande, Moleculenet: a benchmark for molecular machine learning, Chemical science 9 (2) (2018) 513–530.

[46] H. Veith, N. Southall, R. Huang, T. James, D. Fayne, N. Artemenko, M. Shen, J. Inglese, C. P. Austin, D. G. Lloyd, et al., Comprehensive characterization of cytochrome p450 isozyme selectivity across chemical libraries, Nature biotechnology 27 (11) (2009) 1050–1055.

[47] F. Schroff, D. Kalenichenko, J. Philbin, Facenet: A unified embedding for face recognition and clustering, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 815–823. doi:10.1109/CVPR.2015.7298682.

[48] C. Szegedy, S. Ioffe, V. Vanhoucke, A. A. Alemi, Inception-v4, inception-resnet and the impact of residual connections on learning, in: Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI), 2017, pp. 4278–4284.

[49] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, A. Zisserman, Vggface2: A dataset for recognising faces across pose and age, in: 2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG), 2018, pp. 67–74. doi:10.1109/FG.2018.00020.

[50] L. Deng, The mnist database of handwritten digit images for machine learning research [best of the web], IEEE signal processing magazine 29 (6) (2012) 141–142.

[51] H. Xiao, K. Rasul, R. Vollgraf, Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms, arXiv preprint arXiv:1708.07747 (2017).

[52] Y. Wang, J. Xiao, T. O. Suzek, J. Zhang, J. Wang, S. H. Bryant, Pubchem: a public information system for analyzing bioactivities of small molecules, Nucleic acids research 37 (suppl_2) (2009) W623–W633.