



Prototypical clustered federated learning for heart rate prediction^{*#}

Yongjie YIN¹, Hui RUAN¹, Yang CHEN¹, Jiong CHEN², Ziyue LI³,
 Xiang SU⁴, Yipeng ZHOU⁵, Qingyuan GONG^{†#6}

¹School of Computer Science, Fudan University, Shanghai 200438, China

²NIO, Shanghai 201805, China

³Information System Department, University of Cologne, Cologne 50969, Germany

⁴Department of Agricultural Sciences, University of Helsinki, Helsinki 00014, Finland

⁵School of Computing, Macquarie University, Sydney 2109, Australia

⁶Research Institute of Intelligent Complex Systems, Fudan University, Shanghai 200433, China

[†]E-mail: gongqingyuan@fudan.edu.cn

Received Jan. 27, 2025; Revision accepted Apr. 6, 2025; Crosschecked Oct. 16, 2025

Abstract: Predicting future heart rate (HR) not only helps in detecting abnormal heart rhythms but also provides timely support for downstream health monitoring services. Existing methods for HR prediction encounter challenges, especially concerning privacy protection and data heterogeneity. To address these challenges, this paper proposes a novel HR prediction framework, PCFedH, which leverages personalized federated learning and prototypical contrastive learning to achieve stable clustering results and more accurate predictions. PCFedH contains two core modules: a prototypical contrastive learning-based federated clustering module, which characterizes data heterogeneity and enhances HR representation to facilitate more effective clustering, and a two-phase soft clustered federated learning module, which enables personalized performance improvements for each local model based on stable clustering results. Experimental results on two real-world datasets demonstrate the superiority of our approach over state-of-the-art methods, achieving an average reduction of 3.1% in the mean squared error across both datasets. Additionally, we conduct comprehensive experiments to empirically validate the effectiveness of the key components in the proposed method. Among these, the personalization component is identified as the most crucial aspect of our design, indicating its substantial impact on overall performance.

Key words: Federated learning; Heart rate prediction; Prototypical contrastive learning

<https://doi.org/10.1631/FITEE.2500062>

CLC number: TP391

1 Introduction

Heart rate (HR) stands as a crucial health metric used for various applications, including heart disease detection (Patidar et al., 2015), fatigue monitoring (Patel et al., 2011), and activity recognition (Mehrang et al., 2017). Future HR prediction is of significant importance, as it not only aids in detecting HR anomalies through comparison of actual HRs with predicted values but also supports

[‡] Corresponding author

* Project supported by the National Natural Science Foundation of China (Nos. 62102094 and 62072115), the Shanghai Science and Technology Innovation Action Plan Project (No. 22510713600), the NIO University Programme, and the Nordic University Cooperation on Edge Intelligence (No. 168043)

Electronic supplementary materials: The online version of this article (<https://doi.org/10.1631/FITEE.2500062>) contains supplementary materials, which are available to authorized users

ORCID: Qingyuan GONG, <https://orcid.org/0000-0001-7942-8752>

© Zhejiang University Press 2025

downstream detection services such as heart disease and fatigue monitoring, enabling them to provide more timely warnings before the onset of a disease. Extensive research efforts have been dedicated to estimating HR to align with the ground truth values closely. Smart watches (Panwar et al., 2020) and visual methods (Yu et al., 2022) offer more comfortable and non-invasive means of HR estimation compared to alternative health metrics. Consequently, it is feasible and desirable to predict future HR with precise estimation.

Previous studies in this area have heavily depended on a traditional training paradigm, which involves aggregating all HR data and processing the data on a centralized server. This approach raises significant privacy concerns, particularly due to the sensitive nature of HR data. Protecting such data is crucial, especially in light of the General Data Protection Regulation (GDPR), which mandates stringent safeguards for personal data. Federated learning (FL) represents a promising approach for training machine learning models while safeguarding user privacy. This technique allows models to be trained collaboratively across numerous distributed devices, thereby preserving users' privacy while still achieving accurate predictions. Several studies have applied FL for HR prediction and other cardiovascular health problems (Brisimi et al., 2018; Fang et al., 2020; Yaqoob et al., 2023). They often focus on directly identifying heart diseases, which require a substantial amount of user-specific characteristics.

The non-independent and identically distributed (non-IID) problem (Zhu et al., 2021) exists in the FL process due to noticeable differences in data distribution among various clients. It is worth noting that the heterogeneity of HR data exists among individuals. Taking drivers as an example, stressful driving conditions such as heavy traffic or adverse weather conditions may increase a driver's HR. This phenomenon is depicted in Fig. 1, which shows the HR patterns among four drivers during normal driving and stimulated driving. HR fluctuations under stimulation clearly differ from those during normal driving, and there is also variation in the HR performance among different drivers under the same conditions. Consequently, HR responses can vary markedly among individuals, even in similar activity scenarios, which presents a significant non-IID problem. This observation underscores the

inadequacy of applying a uniform predictive model to all clients.

Personalized FL (Tan AZ et al., 2023) has emerged as a primary method to solve the non-IID problem of FL by classifying training samples with similar patterns. In this research, we focus on clustered FL, one type of personalized FL that proves more practical for real-world applications. However, existing clustered FL models only regard the clustering process as a tool to improve personalized training effects, but ignore the importance of achieving better clustering outcomes. Accurate clustering is essential as it ensures that each cluster model is tailored to its specific group, thereby improving the overall performance of cluster prediction models. Furthermore, the clustering process in existing approaches is only beneficial for users who participate in the training process, as it relies on training loss to partition the data, overlooking the users who are temporarily disconnected due to poor network circumstances or high training and transmission costs, which are common in real scenarios.

To address the above challenges, this paper contributes an HR prediction framework named PCFedH, which integrates the advantages of personalized FL and prototypical contrastive learning. PCFedH comprises two fundamental modules: a prototypical contrastive learning-based federated clustering module and a two-phase soft clustered FL module. These modules address data heterogeneity issues in traditional FL from different aspects. The prototypical contrastive learning-based federated clustering module performs feature extraction and clustering on the original HR data to characterize data heterogeneity, enabling the provision of cluster centers for users who did not participate in previous training rounds. The two-phase soft clustered FL module mitigates the instability of clustering performance in existing clustered FL methods, thereby further improving the HR prediction performance of each local model. In the first phase, it initializes clustering and conducts precise data classification based on the results of the prototypical contrastive learning-based federated clustering module. In the second phase, it conducts personalized local training through regularization methods and derives multiple global clustered models based on the weight of data types. Our contributions are threefold.

1. We propose PCFedH, an FL method

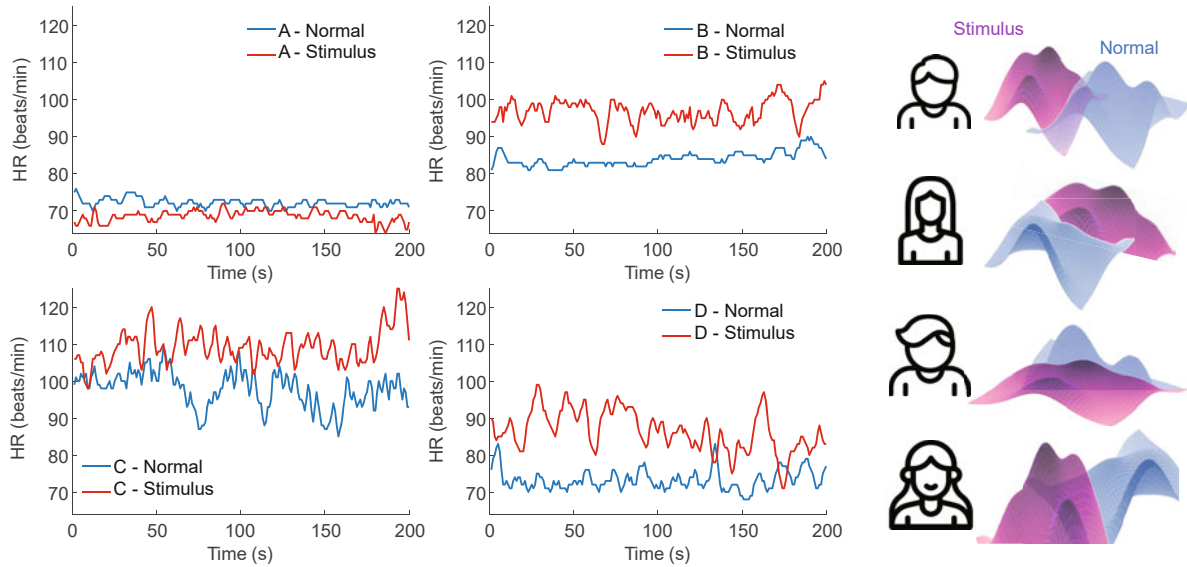


Fig. 1 Heart rate (HR) patterns of four drivers under normal and stimulated conditions. Different individuals exhibit distinct HR patterns and unique inter-state variations under these conditions. For each individual, the two states lie on different manifolds

specifically designed for HR prediction, providing enhanced personalized predictions with practical applicability.

2. To address the non-IID nature of individuals' HR, we synergize prototypical contrastive FL with clustered FL. This approach not only facilitates more stable and improved clustering within FL processes but also leads to superior personalized training outcomes.

3. We evaluate the performance of PCFedH on two real-world datasets, demonstrating significant improvements over baselines. Compared with state-of-the-art methods, our method achieves an average reduction of 3.1% in the mean squared error (MSE), highlighting its effectiveness in accurately predicting HR.

2 Related works

2.1 Heart rate prediction

In the field of HR prediction, most existing works adopt a centralized training paradigm for model training. Research such as Alharbi et al. (2021), Oyeleye et al. (2022), and Staffini et al. (2022) has demonstrated the effectiveness of recurrent neural networks for HR prediction under cen-

tralized training. However, the collection of users' raw data is required, raising significant privacy and security concerns. To address these critical privacy issues, several studies have leveraged FL to predict HR or other cardiac health issues. For example, Fang et al. (2020) proposed an FL-based HR prediction method that relies on real-time exercise speed, which means it is a detection work rather than a prediction issue. Brisimi et al. (2018) developed a federated optimization scheme to train a sparse support vector machine (SVM) for predicting hospitalizations due to heart diseases based on patients' past electronic health records. Yaqoob et al. (2023) applied a federated SVM to identify heart failure by leveraging distinct features of patients' heart diseases. These studies require various user-specific features and fail to account for the fact that each individual's HR exhibits variation characteristics due to factors such as age, behavior, and activity differences. Consequently, training a single unified model cannot meet the diverse needs of all users.

2.2 Clustered federated learning

Clustered FL is one of the core strategies of personalized FL (Sattler et al., 2021). It not only champions privacy by sharing model updates instead of raw data but also allows clients to tailor models

to closely align with their unique data landscapes. The advantage of clustered FL lies in its pragmatic applicability, notably in allowing clients with limited computational resources or new participants to straightforwardly select an appropriate model for task execution based on their data's cluster categorization. This approach is particularly suitable for HR prediction tasks, which involve significant privacy concerns and substantial data heterogeneity across individuals.

IFCA (Ghosh et al., 2022) represents a further evolution in clustered FL, implementing an iterative algorithm that commences with the initialization of K clusters and finally reaches the corresponding global models. Clients are categorized based on the minimal loss function derived from these global models, with each iteration refining the clustering and model personalization process. FedSoft (Ruan and Joe-Wong, 2022) further refines this concept of soft clustering, positing that prior clustered FL models imposed overly stringent constraints on clustering. Acknowledging that client data often span multiple categories, FedSoft advocates for a nuanced model aggregation approach to minimize training overhead. To address the high computational costs brought by model inference through all training data, Cai et al. (2023) introduced FedCE, which is also based on the concept of soft clustering for personalized training. This method uses the loss observed across multiple batches to represent the features of client data distributions, sidestepping the need to specify exact data proportions. However, this approach presumes that each client predominantly belongs to a single class. This assumption may not align with the reality of HR data, where the distribution of data among various categories can be comparably uniform.

In summary, current research on clustered FL merely treats the clustering process as a mechanism to enhance the effects of personalized training, overlooking the significance of achieving accurate clustering outcomes. Moreover, clustering in existing approaches is only useful for users who participate in the training process, leaving non-participating users unable to accurately derive meaningful patterns from HR data.

2.3 Prototype federated learning

The concept of prototypes stems from the domain of prototypical contrastive learning (Li JN

et al., 2021). The contrastive learning approach suggests that each data sample is associated with a specific cluster, with the prototype serving as the central or "average" feature representation of all samples in that cluster. During the learning process, a sample paired with its cluster's prototype forms a positive pair to enhance similarities, while pairing a sample with a prototype from another cluster creates a negative pair to emphasize differences.

In the realm of FL, the integration of prototypes into learning models has been implemented in various ways (Tan Y et al., 2022a, 2022b; Mu et al., 2023; Qi et al., 2023), primarily within supervised learning frameworks. For example, FedProto (Tan Y et al., 2022b) uses a paradigm in which clients share knowledge with each other through their prototypes rather than gradients, while FedPCL (Tan Y et al., 2022a) uses local and global prototypes. However, by focusing primarily on supervised tasks, especially classification, these methods inadvertently narrow the scope of prototype utilization. This limitation confines their utility and represents a missed opportunity for broader application in unsupervised or semi-supervised learning scenarios, where prototypes could offer substantial benefits in learning from unlabeled or partially labeled data.

In this work, we bridge this critical gap by combining prototype FL and clustered FL. We employ clustering techniques to generate cluster centers from unlabeled HR data and apply prototype FL to enhance the predictors' abilities. These can not only facilitate the training of high-quality global encoders with low additional communication costs but also expand the application of prototype FL to unsupervised tasks.

3 Method

3.1 Problem definition

Leveraging FL in HR prediction typically involves a central server and N clients within a horizontal FL setting, where these clients possess local HR datasets $\{D_1, D_2, \dots, D_N\}$. Here, $D_i = \{\mathbf{X}_{i,h}, Y_{i,h}\}$ ($h = 1, 2, 3, \dots, n_i$), with n_i representing the number of data samples across the i^{th} client. The central server is responsible for model aggregation, while the vehicles, serving as clients, participate in FL to process HR data and train models. The model input

is a univariate time series of HR values measured at L consecutive time points $\mathbf{X} = (x_1, x_2, \dots, x_L)$ with $x_i \in \mathbb{R}$. The prediction target of the model is the future HR value at the next time point $Y = y_{L+1} \in \mathbb{R}$ following \mathbf{X} . Each client trains its local prediction model (with model parameters denoted as \mathbf{w}_i) to tackle this regression problem using its local dataset D_i . We define the HR prediction problem under the traditional FL scenario, and then proceed to analyze the problem using clustered FL models.

In traditional FL, the parameters of each local prediction model are identical before the local training begins, having been downloaded from the server to overwrite the local model. The objective function for each local training is identical, denoted as \mathcal{L}_P . Hence, the overall objective function of FL is obtained by

$$\min \sum_{i=1}^N \frac{n_i}{n} \mathcal{L}_i(\mathbf{w}_i, D_i), \quad (1)$$

where \mathcal{L}_i denotes the local objective function of the i^{th} client. The initial conditions are identical across all clients, satisfying $\mathcal{L}_P = \mathcal{L}_1 = \mathcal{L}_2 = \dots = \mathcal{L}_N$ and $\mathbf{w}_p = \mathbf{w}_1 = \mathbf{w}_2 = \dots = \mathbf{w}_N$. Traditional FL methods synchronize models across all clients by periodically distributing the global model from the server. However, these parameter synchronization approaches may bias the model towards samples in greater quantities, potentially degrading performance on local datasets in the face of data heterogeneity among clients.

Next, we detail the specific problem definition where the heterogeneity in HR is considered. Given the variability of HR data, we assume the local dataset comprises data following various distributions; i.e., each data pair $\{\mathbf{X}, Y\}$ is drawn from one of K distributions P_1, P_2, \dots, P_K . The proportion of different distributions varies across each client's local dataset, making the unified synchronization strategy used in traditional FL inapplicable. Therefore, we adopt the concept of clustered FL to define the problem, aiming to enhance each client's predictive performance through personalized prediction. Unlike traditional FL, which relies on a single global model, clustered FL aims to train global model parameters $\mathbf{w}_{g,1}, \mathbf{w}_{g,2}, \dots, \mathbf{w}_{g,K}$ for K distributions, with each parameter set $\mathbf{w}_{g,k}$ ($k = 1, 2, \dots, K$) corresponding to a specific data distribution P_k and optimized for its performance. The training of multiple clus-

tered global models also depends on the knowledge of the HR data distribution across clients, which is determined by users' activity conditions and personal states and is often not provided in real-world applications. Thus, in our method, we employ clustering methods to effectively address the challenge of data heterogeneity before the clustered FL module.

In summary, we aim to train HR prediction models through personalized FL, intending to preserve user privacy while using historical HR data to predict future HRs. The solution covers two main aspects: (1) high-quality clustering of local HR datasets to characterize data heterogeneity across clients; (2) implementation of cluster personalized FL based on clustering results to address data heterogeneity issues.

We achieve personalized training through the concept of clustered FL, requiring the server to update a set of K cluster model parameters $\mathbf{w}_{g,k}$. Each client's local model parameters \mathbf{w}_i are trained using the K cluster models combined with auxiliary information about the number of data samples n_{ik} from each distribution in its local data. The overall objective function for personalized FL is obtained by

$$\min \sum_{i=1}^N \mathcal{L}_i(\mathbf{w}_i, \mathbf{w}_{g,k}, n_{ik}), \quad (2)$$

where \mathcal{L}_i is the local objective function with individual differences in local objective functions among clients.

3.2 PCFedH architecture

Fig.2 presents the overall architecture of PCFedH, primarily comprising two core modules: a prototypical contrastive learning-based federated clustering module and a two-phase soft clustered FL module. In the process of personalized federated training, the effectiveness of local model updates depends critically on n_{ik} , which itself is determined by the effectiveness of data clustering. The prototypical contrastive learning-based federated clustering module is responsible for achieving privacy preserving-clustering in each client. Each client trains its cluster representation encoder \mathbf{w}_i^e on local data, and performs clustering on the encoded features to generate cluster centers C_i . The server is responsible for aggregating local cluster centers to obtain global cluster centers $C_g = \{c_1, c_2, \dots, c_K\}$, where c_k denotes the

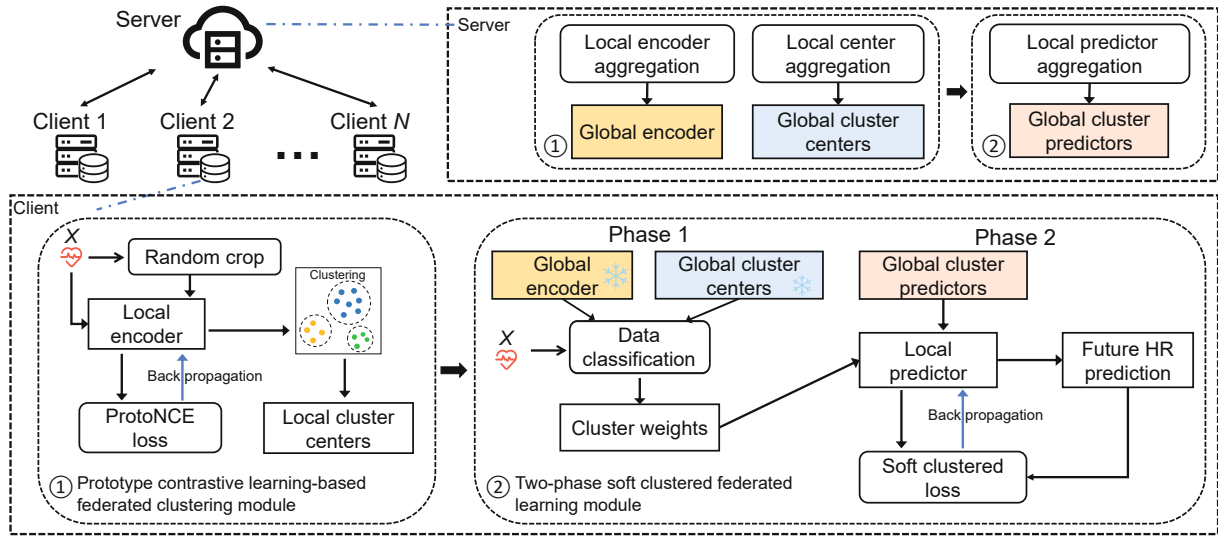


Fig. 2 Architecture of PCFedH consists of two core modules. The first module details the process where clients train the clustering encoder on their local data, perform local clustering, and then upload the local cluster centers and local model to the server for the aggregation of a global encoder and global cluster centers. The second module describes how clients use the global cluster centers and global encoder to cluster their local data and calculate cluster weights (Phase 1), and then train their local prediction models with the soft clustered FL approach to improve prediction accuracy on their datasets and contribute to the enhancement of the global cluster prediction models (Phase 2). Notably, the second module is dependent on the global encoder and cluster centers generated in the first module

cluster center corresponding to the k^{th} data distribution. It then combines local encoders into a global cluster representation encoder w_g^e . This process iterates until the clustering results converge. Section 3.3 introduces the detailed design of this module.

The two-phase soft clustered FL module is responsible for training personalized FL prediction models based on clustering results. In the first phase, data classification is first performed. Each client uses the final global cluster centers C_g and the global encoder w_g^e obtained from the prototypical contrastive learning-based federated clustering module to cluster its data, generating different category weights. These weights refer to the proportion of data belonging to various clustering categories relative to the total dataset, denoted as u_{ik} . In the second phase, using these weights, each client trains its local prediction model w_i with personalized FL methods. This allows clients to achieve improved prediction results on their respective datasets while simultaneously enhancing the performance of global cluster prediction models $w_{g,k}^p$, especially improving the performance of models related to categories that are more prevalent in their data distributions. Section 3.4 introduces the detailed design of this module.

3.3 Prototypical contrastive learning-based federated clustering

We design the prototypical contrastive learning-based federated clustering module for feature extraction. This module comprises two parts: prototypical contrastive learning, which constructs positive and negative sample pairs using original representations and cluster centers, and contrastive learning through contextual consistency.

Algorithm 1 details the prototypical contrastive learning-based federated clustering. The server initializes global cluster centers C_g and a global cluster representation encoder model w_g^e (line 1 in the main body of Algorithm 1). In each FL round, each client downloads the latest global encoder from the server and updates its local encoder. Subsequently, the local encoder is trained for a contrastive learning task (lines 1–6 in the ClientUpdate function of Algorithm 1). Specifically, representation extraction is performed, converting the original HR time series X_h into an encoded feature vector $z_h \in \mathbb{R}^q$ (q represents the dimension of encoded feature vectors) using the local encoder.

Prototypical contrastive learning is employed to

Algorithm 1 Prototypical contrastive learning-based federated clustering

Input: Local HR dataset D_i from each client, $i = 1, 2, \dots, N$, the number of samples across the i^{th} client n_i , and $n = \sum_{i=1}^N n_i$.

Server execution:

- 1: Initialize the global encoder \mathbf{w}_g^e and global cluster centers C_g
- 2: **for** each round $t = 1, 2, \dots, R_{\text{pcl}}$ **do**
- 3: Send \mathbf{w}_g^e and C_g to all clients
- 4: **for** each client i in parallel **do**
- 5: $\mathbf{w}_i^e, C_i \leftarrow \text{ClientUpdate}(D_i, \mathbf{w}_g^e, C_g)$
- 6: **end for**
- 7: $\mathbf{w}_g^e \leftarrow \sum_{i=1}^N \frac{n_i}{n} \mathbf{w}_i^e$ /* Aggregate local encoders */
- 8: $C_g \leftarrow \sum_{i=1}^N \frac{1}{N} C_i$ /* Aggregate local cluster centers */
- 9: **end for**

ClientUpdate (D_i, \mathbf{w}_g^e, C_g):

- 1: **for** each local epoch $\tau = 1, 2, \dots, E$ **do**
 - 2: **for** each batch b in D_i **do**
 - 3: Calculate $\mathcal{L}_{\text{ProtoNCE}}$ for b
 - 4: Update the encoder: $\mathbf{w}_i^e \leftarrow \mathbf{w}_g^e - \eta \nabla \mathcal{L}_{\text{ProtoNCE}}(b, \mathbf{w}_g^e)$
/* η denotes the encoder learning rate for each client */
 - 5: **end for**
 - 6: **end for**
 - 7: Perform one step of local K -means clustering to update C_i
 - 8: **return** \mathbf{w}_i^e and C_i to the server
-

enhance clustering quality. In this approach, each sample and its corresponding cluster center form a positive pair, while negative samples are formed with other cluster centers. By contrasting each sample with prototypes of its own and other categories, the goal is to maximize intra-class similarity and minimize inter-class similarity (Li JN et al., 2021). Features obtained through prototypical contrastive learning demonstrate robustness to noise and are naturally suited for clustering tasks, making clustering more precise and meaningful. The loss function for this learning process is as follows:

$$\mathcal{L}_{\text{pcl}} = \sum_{h=1}^{n_i} -\log \frac{\exp(\mathbf{z}_h \cdot \mathbf{c}_p / \phi_p)}{\sum_{k=1}^K \exp(\mathbf{z}_h \cdot \mathbf{c}_k / \phi_k)}, \quad (3)$$

where ϕ is a class density value controlling the gap between features and each cluster center, obtained through the clustering density of cluster centers. \mathbf{c}_p is the cluster center belonging to the class of the

encoded feature vector \mathbf{z}_h , determined by calculating the cosine similarity:

$$p = \underset{k}{\operatorname{argmax}} \operatorname{CosSim}(\mathbf{z}_h, \mathbf{c}_k). \quad (4)$$

To ensure a more stable and efficient convergence of the encoder training process, the training includes contrastive learning through contextual consistency. This method is particularly suited for time series data, mitigating the impact of outliers and abrupt changes (Yue et al., 2022). The process of constructing positive and negative sample pairs through contextual consistency includes two parts: random cropping and random masking. Random cropping selects two subsequences, denoted as $[a_1, b_1]$ and $[a_2, b_2]$, from an original input sequence of length L . These subsequences contain repeated segments and are randomly chosen to satisfy the condition $0 < a_1 < a_2 < b_1 < b_2 \leq L$. The second part, random masking, involves applying masking operations to the two previously selected subsequences. This process follows a Bernoulli distribution where each temporal variable has an equal probability of being zeroed out. Specifically, the probability is set to 50% for every variable at each time point.

By processing the two subsequences in this manner, they are likely to retain similar contextual information, while random masking enhances robustness to abrupt changes and outliers. The feature vectors \mathbf{f}_h and \mathbf{f}_h' obtained by encoding these two subsequences are treated as positive sample pairs, while the feature vector pairs of other subsequences within the same training batch are treated as negative sample pairs. The loss function for contextual consistency contrastive learning, denoted as $\mathcal{L}_{\text{InfoNCE}}$, is formulated as follows:

$$\mathcal{L}_{\text{InfoNCE}} = \sum_{h=1}^{n_i} -\log \frac{\exp(\mathbf{f}_h \cdot \mathbf{f}_h')}{\sum_{j=1}^r \exp(\mathbf{f}_h \cdot \mathbf{f}_j')}, \quad (5)$$

where \mathbf{f}_h' is the positive sample corresponding to \mathbf{f}_h , and r is the number of negative samples, which depends on the size of the training batch.

The goal of each local training session is to minimize a combined loss function consisting of two terms, defined as

$$\mathcal{L}_{\text{ProtoNCE}} = \mathcal{L}_{\text{pcl}} + \alpha \mathcal{L}_{\text{InfoNCE}}, \quad (6)$$

where α is a weighting coefficient.

After completing local contrastive learning training, each client uses its local encoder to extract features from its local data and subsequently performs clustering using the K -means algorithm (Jain and Dubes, 1988). In the federated clustering process designed in this paper, clustering iterations and the entire FL iteration process occur simultaneously. Specifically, after each round of FL, each client uses the global cluster centers $C_g = \{c_1, c_2, \dots, c_K\}$ provided by the server as the initial cluster centers rather than performing random initialization. For the encoded feature z_h of each client's data point x_h after local training, the Euclidean distance $d(z_h, c_k)$ to each cluster center c_k is calculated. The data point x_h is assigned to the cluster with the nearest distance, thus dividing the local data into K clusters, expressed as

$$\text{Cluster}(x_h) = \underset{k}{\operatorname{argmin}} d(z_h, c_k), \quad (7)$$

where $\text{Cluster}(x_h)$ indicates the cluster assignment of x_h . For each cluster k , the new local cluster center c'_k is computed as the mean of the encoded features for all data points within the cluster:

$$c'_k = \frac{1}{|S_k|} \sum_{z_h \in S_k} z_h, \quad (8)$$

where S_k denotes the set of data points assigned to the k^{th} cluster and $|S_k|$ represents the number of data points in that set. The key aspect of the federated clustering process is that each client performs clustering of local data and updates the new cluster centers based on the global cluster centers (line 7 in the ClientUpdate function of Algorithm 1). This approach significantly accelerates computation while mitigating the risk of cluster centers overfitting caused by incomplete data categories at the local level. After generating new local cluster centers, clients upload the local cluster centers and local encoders to the server. The server aggregates these local cluster centers and local encoders to update the global cluster centers and global encoder (lines 7–8 in the main body of Algorithm 1). This iterative process—comprising local training, cluster center updating, and server-side aggregation—continues until the global encoder achieves satisfactory performance or the predetermined number of FL rounds is reached.

3.4 Two-phase soft clustered federated learning

We use FedSoft (Ruan and Joe-Wong, 2022), a state-of-the-art clustered FL technique, to train the HR prediction model, addressing the non-IID challenge. FedSoft's uniqueness lies in its capability to efficiently train both clustered and personalized models. Recognizing that a client might possess diverse data types, FedSoft employs a proximal approach to derive multiple cluster models without multiple training processes. It categorizes data based on training loss using distinct clustering methods and employs an expectation-maximization (EM) algorithm to ensure the convergence of clustering and prediction processes. However, the EM algorithm has its limitations. It can be plagued by slow convergence and might settle at local maxima instead of global ones. Furthermore, for new users to use the cluster model, they must first classify their data, a step not included in FedSoft. To address FedSoft's limitations, we eliminate the EM process and instead cluster the data initially.

This module effectively addresses the challenge of non-IID data in FL through precise data classification and personalized training based on category weights. The accurate classification in the first phase provides a stable foundation for subsequent personalized training, which optimizes the model specifically for the data distribution of each client, thereby enhancing the overall model performance and adaptability. The details of the two phases are shown in Algorithm 2, which will be discussed below.

1. Precise data classification. This phase addresses the instability in clustering caused by an undertrained initial global model. The focus of this phase is to establish a stable and reliable data classification foundation for each client, setting the stage for personalized model training. We use the global cluster representation encoder and cluster centers from the prototypical contrastive learning-based federated clustering module for accurate data classification. By calculating the cosine similarity between each data representation and the cluster centers, we compute the number of data points belonging to every cluster k for client i , denoted as n_{ik} . Subsequently, for each client, the proportion of each type of data is used as the weight for different categories:

$$u_{ik} = \frac{n_{ik}}{n_i}. \quad (9)$$

Algorithm 2 Two-phase soft clustered federated learning for heart rate prediction

Input: The local HR dataset of each client D_i for $i = 1, 2, \dots, N$, a pretrained global encoder \mathbf{w}_g^e , and cluster centers $C_g = \{\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_K\}$

Server execution:

```

1: // Phase 1: Data categorization
2: Broadcast  $\mathbf{w}_g^e$  and  $C_g$  to all clients
3: for each client  $i$  in parallel do
4:   for each data point  $\mathbf{X}_{i,h}$  do
5:      $j = \underset{k}{\operatorname{argmax}} \operatorname{CosSim}(\mathbf{c}_k, \mathbf{w}_g^e(\mathbf{X}_{i,h}))$ 
6:      $n_{ij} = n_{ij} + 1$ 
7:   end for
8:   Calculate  $u_{ik}$  according to Eq. (9)
9:   return  $n_{ik}$  to the server
10: end for
11: The server calculates  $v_{ik}$  according to Eq. (10)
12: // Phase 2: Personalized FL
13: Initialize global cluster predictors  $\mathbf{w}_{g,k}^p, k = 1, 2, \dots, K$ 
14: for each round  $t = 1, 2, \dots, R_{\text{scl}}$  do
15:   The server randomly selects  $K$  sets of clients  $\text{Sel}_k \subset \{1, 2, \dots, N\}$ , where  $|\text{Sel}_k| = M$ , and each client is selected with probability  $v_{ik}$ 
16:   for each selected client  $i$  in parallel do
17:     Download  $\mathbf{w}_{g,k}^p, k = 1, 2, \dots, K$ 
18:      $\mathbf{w}_i^p \leftarrow \text{ClientTrainPredict}(D_i, \mathbf{w}_{g,k}^p, n_{ik})$ 
19:   end for
20:   Update  $\mathbf{w}_{g,k}^p \leftarrow \frac{1}{M} \sum_{i \in \text{Sel}_k} \mathbf{w}_i^p$ 
21: end for
ClientTrainPredict( $D_i, \mathbf{w}_{g,k}^p, n_{ik}$ ):
1: for each local epoch  $\tau = 1, 2, \dots, E$  do
2:   for each batch  $b$  in dataset  $D_i$  do
3:     Compute  $\mathcal{L}_i(b, \mathbf{w}_i^p, \mathbf{w}_{g,k}^p, n_{ik})$ 
4:     Update the local predictor:  $\mathbf{w}_i^p \leftarrow \mathbf{w}_i - \eta \nabla \mathcal{L}_i$ 
5:   end for
6: end for
7: return  $\mathbf{w}_i^p$  to the server

```

Then, the server computes the aggregation weight v_{ik} for client i as follows:

$$v_{ik} = \frac{n_{ik}}{\sum_i n_{ik}}. \quad (10)$$

2. Personalized FL. Building on the classification outcomes from the first phase, the second phase focuses on enhancing the adaptability and prediction accuracy of each client's model towards its local data through personalization. To simulate real-world scenarios and reduce the high costs of training and transmission, only a subset of clients is selected to participate in each training round. Specifi-

cally, the server initially selects K groups of clients, $\text{Sel}_k \subset \{1, 2, \dots, N\}$, based on the probability v_{ik} (line 15 in Algorithm 2), where $|\text{Sel}_k| = M < N$, and M is a predefined hyperparameter denoting the number of clients per group. This selection means that the M clients in Sel_k are likely to contain more class k data. These K groups of clients correspond to the aggregation and update of the K global cluster predictors. Each client downloads the global clustering prediction model $\mathbf{w}_{g,k}^p$ and adjusts the regularization term in its loss function based on the weight of the data category. The proximal update for each client's local prediction model is performed as follows:

$$\mathcal{L}_P(\mathbf{w}_i^p) = \frac{1}{n_i} \sum_{h=1}^{n_i} (\mathbf{w}_i^p(\mathbf{X}_{i,h}) - Y_{i,h})^2, \quad (11)$$

$$\mathcal{L}_i(\mathbf{w}_i^p) = \mathcal{L}_P(\mathbf{w}_i^p) + \frac{\lambda}{2} \sum_{k=1}^K u_{ik} \|\mathbf{w}_i^p - \mathbf{w}_{g,k}^p\|^2, \quad (12)$$

where \mathbf{w}_i^p represents the parameters of the local prediction model and K is the number of cluster centers. In Eq. (12), $\mathcal{L}_P(\mathbf{w}_i^p)$ represents the primary loss function, which in this case is the MSE. The second term $\frac{\lambda}{2} \sum_{k=1}^K u_{ik} \|\mathbf{w}_i^p - \mathbf{w}_{g,k}^p\|^2$ serves as a regularization term. It is weighted by each client's data category distribution u_{ik} , and the distance is calculated between the model parameters \mathbf{w}_i^p and each global cluster model parameter $\mathbf{w}_{g,k}^p$. This mechanism retains the personalized features of the local prediction model and ensures the model is as close as possible to the model corresponding to the dominant category in the local dataset. Here, λ is a regularization parameter that balances the two terms.

After training, the clients upload their local models to the server. The server then aggregates each global cluster prediction model for updates. Specifically, the global cluster model $\mathbf{w}_{g,k}^p$ is updated through aggregation based solely on the local prediction models of the clients in the group Sel_k , which is selected by the probability based on the category proportion (line 20 in Algorithm 2). In this manner, each client's local prediction model is tailored to its specific dataset and encouraged to align closely with the global class prediction model corresponding to the dominant category in the client's data.

3.5 Computational complexity analysis

In this subsection, we compare our method with other FL methods in terms of computational

complexity, including classical FL methods (e.g., FedAvg (McMahan et al., 2017), FedProx (Li T et al., 2020), and FedALA (Zhang et al., 2023)), clustered FL methods (e.g., IFCA (Ghosh et al., 2022), FedSoft (Ruan and Joe-Wong, 2022), and FedCE (Cai et al., 2023)), and prototype FL methods (e.g., FedPCL (Tan Y et al., 2022a) and FedProto (Tan Y et al., 2022b)).

Here, we focus on the time complexity of one training epoch on the client side, as the computational resources of the central server are typically sufficient. Experimental results are presented in Table 1.

In this comparison, we consider the computational costs of all operations and omit the specific prediction or classification model employed on the client side. The following basic notations are used: d denotes the dimension of models, B the dataset size, K the number of clusters, m the number of clients, and p a special hyperparameter of FedALA. As shown in Table 1, the time complexity of PCFedH, our proposed FL framework, is comparable to or even greater than other state-of-the-art clustered and prototype FL methods. This indicates that the computational cost of PCFedH on the client side is acceptable. Meanwhile, PCFedH exhibits higher time complexity than classical FL methods (e.g., FedAvg), which rely on simpler client-side operations. This increase stems from its additional components (e.g., prototype contrast and clustering) designed to enhance personalized prediction performance and robustness in non-IID scenarios. These advantages are empirically validated in Section 5.

4 Convergence analysis

In this section, we provide the theoretical convergence guarantees for PCFedH. Since the two modules are trained sequentially and independently in our work, we prove their convergence separately in two parts. Specifically, we first prove the conver-

gence of module 1, where the upper bound of its gradient tends to zero during the training. Second, we analyze the convergence of module 2, where the gap between its global loss and optimal loss tends steadily to zero. For clear illustration and easy writing, we inherit the notation system from Algorithms 1 and 2.

4.1 Convergence of module 1

We provide insights into the convergence analysis of PCFedH, as demonstrated in Theorem 1. A detailed description and derivations can be found in the supplementary materials.

Theorem 1 Combining basic assumptions of smoothness, bounded variance, and dissimilarity, for a sufficiently small learning rate $\eta \leq \sqrt{\frac{N}{TE}}$, it can be proven that the parameters of module 1 converge during the training process

$$\min_{t \in \{1, 2, \dots, T\}} \mathbb{E} \left[\|\nabla F(\mathbf{w}_{t,0}, \mathbf{C}_{t,0})\|^2 \right] \leq \mathcal{O} \left(\frac{1}{\sqrt{NET}} \right) + \mathcal{O} \left(\frac{EL_F \sigma^2}{n^2} \sqrt{\frac{NE}{T}} \sum_{i=1}^N n_i^2 \right) + \mathcal{O} \left((\sigma^2 + \kappa^2 E) \frac{NEL_F^2}{T} \right),$$

where $\|\cdot\|$ represents the Euclidean norm, $\mathbb{E}[\cdot]$ denotes the expectation with respect to the randomness of data sampling or client selection, and $F(\cdot)$ denotes the global objective function. T denotes the number of training rounds. E represents the number of local training epochs for each client, σ indicates the upper bound of the variance of stochastic gradients during local updates, and κ is a constant satisfying $\kappa^2 \geq 0$. L_F denotes the smoothness constant, implying that each local objective function is L_F -smooth. The detailed proof is shown in the supplementary materials.

4.2 Convergence of module 2

Module 2 is designed to train efficient personalized local HR predictors based on representative prototypes obtained from module 1. The framework of module 2 is similar to FedSoft, whose convergence

Table 1 Computational complexity comparison between our methods and baselines

Method	FedAvg	FedProx	FedALA	IFCA	FedSoft	FedCE	FedPCL	FedProto	PCFedH (M1)	PCFedH (M2)
Time complexity	$\mathcal{O}(Bd)$	$\mathcal{O}(Bd)$	$\mathcal{O}(B(d+p))$	$\mathcal{O}(BKd)$	$\mathcal{O}(BKd)$	$\mathcal{O}(BKd \log K)$	$\mathcal{O}(BKdM)$	$\mathcal{O}(BKd)$	$\mathcal{O}(BKd)$	$\mathcal{O}((B+K)d)$

B : dataset size; d : model dimension; K : cluster count; M : client count; p : a special hyperparameter of FedALA; M1/M2: our framework's two core modules. $\mathcal{O}(\cdot)$ denotes the asymptotic computational complexity of algorithms

has been completely proven by Ruan and Joe-Wong (2022). Structurally differing from FedSoft in major aspects, module 2 performs data classification once based on reliable representative prototypes to reduce computational costs. These differences do not affect the convergence of the framework.

5 Experiments

5.1 Experimental settings

5.1.1 Data preparation

We evaluate the performance of PCFedH based on two representative HR datasets.

1. PPG-DaLiA (Reiss et al., 2019): a real-world dataset containing HR data from the daily life activities of 15 subjects. It has been used in several research studies, including one by Xu et al. (2022). We use the data to generate 15 local HR datasets.

2. Distracted (Taamneh et al., 2017): a real-world dataset from 68 volunteers (with 63 effective samples) in a simulated driving environment. It has been used in several research studies, including one by Chen et al. (2021).

We use data from both normal and distracted driving conditions to generate 63 local HR datasets. Each HR series in the dataset originates from a distinct individual, with each individual assigned to a separate client in our experimental setup. The configuration prevents the sharing of data from the same individual across clients. Thus, the heterogeneity of HR data among individuals can be maintained. Additionally, we set the input HR series length L to 150, where each continuous subseries is treated as an input, with the subsequent time scale data serving as the prediction target. For the training phase, 80% of the input-output pairs are used, while the remaining 20% are used for testing.

5.1.2 Implementation details

Following Yue et al. (2022), the encoder in the prototypical contrastive learning-based federated clustering module is a TS2Vec model, which consists of an input projection layer that maps time series observations to high-dimensional latent vectors, a timestamp masking module for creating augmented context views by randomly masking selected timestamps, and a dilated convolutional neural network

module with 10 residual blocks to extract contextual representations at each timestamp. The prediction model in the two-phase soft clustered FL module consists of a single-layer long short-term memory module followed by a fully connected layer.

We select Adam (Kingma and Ba, 2015) as the optimizer with a learning rate $\eta = 0.01$. The local batch size is set to 16. Each client performs $E = 5$ local updates per round in both modules. The contrastive learning loss function uses a weight coefficient $\alpha = 1$ and regularization parameter $\lambda = 0.001$. Regarding the clustering configuration, we set the number of classes to $K = 4$ for the PPG-DaLiA dataset and $K = 5$ for the Distracted dataset. Additionally, the number of selected clients is set to $M = 5$ for the PPG-DaLiA dataset and $M = 20$ for the Distracted dataset. The prototypical contrastive learning-based federated clustering module runs $R_{\text{pcl}} = 10$ rounds, while the two-phase soft clustered federated learning module runs $R_{\text{scl}} = 500$ rounds. We implement all the methods using PyTorch and conduct all experiments on two NVIDIA Quadro P5000 GPUs. We use the following three metrics for evaluation: MSE, the mean absolute error (MAE), and the mean absolute percentage error (MAPE), which are widely used across various fields (Rather et al., 2015; Yao et al., 2019).

5.2 Experimental results

5.2.1 Overall performance

We evaluate the performance of PCFedH on the HR prediction task by comparing it against several FL methods in the main experiment. The traditional FL baseline methods include FedAvg, FedProx, and FedALA. As the foundational framework, FedAvg was the first to create the method of combining model update gradients across different devices. Building upon this, FedProx introduced a special proximal term to tackle the statistical heterogeneity inherent in FL. FedALA further advanced the paradigm through adaptively aggregating the global model and local model towards the local objective on each client. Moreover, we compare our method with several state-of-the-art personalized FL methods, including clustered FL and prototype FL, which are introduced in Section 2.

Table 2 presents the experiment results. The proposed method PCFedH outperforms all the

Table 2 Performance comparison of PCFedH and baseline methods on the datasets of PPG-DaLiA and Distracted

Method	PPG-DaLiA			Distracted		
	MSE	MAE	MAPE (%)	MSE	MAE	MAPE (%)
FedAvg	2.621±2.060	1.121±0.398	1.445±0.583	20.242±153.889	1.466±4.076	1.486±1.804
FedProx	<u>2.597±1.784</u>	1.126±0.388	<u>1.441±0.553</u>	<u>1.665±1.209</u>	<u>0.929±0.393</u>	<u>1.202±0.519</u>
FedALA	2.615±2.029	1.121±0.396	1.445±0.581	46.470±13.783	6.610±0.898	8.819±2.525
IFCA	2.624±2.008	1.125±0.393	1.449±0.575	19.868±151.646	1.404±4.018	1.402±1.773
FedSoft	2.737±1.861	1.155±0.394	1.478±0.556	1.671±1.214	0.930±0.395	1.205±0.522
FedCE	2.610±1.987	<u>1.121±0.391</u>	1.443±0.573	18.649±140.433	1.473±3.870	1.532±1.716
FedProto	23.133±19.097	3.559±1.552	4.445±1.658	99.373±450.702	4.503±7.050	4.715±3.565
FedPCL	11.239±24.140	2.410±2.330	3.028±2.996	38.898±244.490	3.532±5.141	4.144±3.996
PCFedH	2.549±1.736	1.117±0.383	1.431±0.549	1.593±1.175	0.903±0.398	1.170±0.526

MAE: mean absolute error; MAPE: mean absolute percentage error; MSE: mean squared error. The best result is highlighted in boldface, and the second-best result is underlined. It shows PCFedH's superior results in reducing MSE, MAE, and MAPE on the PPG-DaLiA and Distracted datasets

baselines across both datasets. It achieves average reductions of 3.1% in MSE, 1.6% in MAE, and 1.7% in MAPE compared with the second-best results, demonstrating its superior ability to predict HR while preserving privacy. Additionally, through comparisons of various methods, we have reached the following two conclusions:

1. The robustness of regularized local updates. Considering that FedSoft, FedProx, and PCFedH all employ regularization strategies during local updates, we can observe that methods based on regularized updates perform well in both datasets. This indicates that for HR time series data, regularized updates have better robustness compared to overwriting updates.

2. The importance of high-quality clustering for personalized clustered FL. Clustered FL aims to enhance the model's personalization capabilities by grouping similar clients.

The effectiveness of clustering directly affects the quality of personalization. Poor clustering results may lead to the model's inability to correctly identify and adapt to the specificity of client data, thus limiting the model's personalized performance. As shown by the performance of FedSoft and IFCA, merely applying clustering algorithms does not guarantee performance improvement. Similarly, approaches like FedProto and FedPCL, which share prototypes without clustering, fail to achieve accurate predictive outcomes. In contrast, the excellent performance of PCFedH on two datasets emphasizes the importance of an efficient clustering mech-

anism for improving the personalization capabilities of FL systems. This indicates that PCFedH can more effectively recognize and use inter-client differences, enhancing the model's personalized adaptability through precise clustering and successfully expanding the prototype technology into unsupervised time series pattern classification.

5.2.2 Ablation study

In this subsection, we present the results of ablation experiments to evaluate the effectiveness of the four key components: prototypical contrastive learning (PCL), cluster representation encoder (CRE), the data classification phase (DCP), and soft clustered (SC) FL. The ablation results are shown in Table 3.

Prototypical contrastive learning: To underscore the significance of prototypical contrastive learning, we compare PCFedH with a variant (denoted as PCFedH (w/o PCL)) that only uses the sub-series contrastive learning to train the representation encoder. The comparison highlights the essential role of prototypical contrastive learning, which significantly improves the prediction ability of the model. This enhancement stems from the ability of prototypical contrastive learning to better align data with their cluster center, benefiting clustering outcomes and consequently improving the prediction performance.

Cluster representation encoder: The importance of the cluster representation encoder is

Table 3 Ablation study: SoftCluster is the most important design

Method	MSE	MAE	MAPE (%)
PCFedH (w/o PCL)	<u>2.553±1.729</u>	<u>1.118±0.381</u>	1.431±0.542
PCFedH (w/o CRE)	2.565±1.762	1.119±0.381	1.432±0.543
PCFedH (w/o DCP)	2.737±1.861	1.155±0.394	1.478±0.556
PCFedH (w/o SC)	2.785±1.957	1.165±0.379	1.490±0.546
PCFedH	2.549±1.736	1.117±0.383	<u>1.431±0.549</u>

MAE: mean absolute error; MAPE: mean absolute percentage error; MSE: mean squared error; w/o: without; PCL: prototypical contrastive learning; CRE: cluster representation encoder; DCP: data classification phase; SC: soft clustered. The best result is highlighted in boldface, and the second-best result is underlined

demonstrated by comparing PCFedH to a variant (denoted as PCFedH (w/o CRE)) that employs only original time series data for clustering. This comparison validates the necessity of the cluster representation encoder, as it learns deeper representations that yield improved clustering results.

Data classification phase: To illustrate the significance of the data classification phase, we examine a variant (denoted as PCFedH (w/o DCP)) of PCFedH that follows the same approach as FedSoft. The noticeable decline in performance with this variant indicates that relying solely on the EM process for clustering limits the efficacy of clustering and prediction outcomes. This finding validates the importance of the data classification phase, which provides superior clustering performance and enhances clustered FL outcomes.

Soft clustered FL: The value of soft clustered FL is demonstrated through a comparison with a variant (denoted as PCFedH (w/o SC)) of PCFedH that omits the SoftCluster component. In this variant, data are initially segmented and categorized using the fast prototypical federated contrastive learning module, after which unique global models are independently trained for each data type. The marked decline in this variant's performance suggests that different data types can still mutually benefit each other. This highlights the necessity of soft clustered

FL, as it allows clients to learn personalized local models without the constraints of hard clustering.

5.2.3 Hyperparameter sensitivity analysis

We investigate the sensitivity of the learning round R_{pcl} within the two-phase soft clustered FL module, with our findings detailed in Table 4. By selecting an optimal value for R_{pcl} , we can achieve a balance between performance and computational cost. The results indicate that setting $R_{pcl} = 10$ yields satisfactory performance. Considering the negligible gains observed when increasing R_{pcl} from 10 to 100, we opt to set $R_{pcl} = 10$ for PCFedH to maintain efficiency without sacrificing accuracy.

In summary, setting $R_{pcl} = 10$ in the two-phase soft clustered FL module achieves satisfactory performance, maintaining efficiency without significant sacrifice in accuracy for PCFedH.

5.2.4 Effectiveness of cluster representation

To illustrate the effectiveness of our cluster representation, we employ *t*-SNE (van der Maaten and Hinton, 2008), a technique primarily used for feature dimensionality reduction to simplify cluster identification. We use this method to perform dimensionality reduction on the representations of the PPG-DaLiA dataset, projecting them into a two-dimensional space for visualization, as shown in

Table 4 Performance of PCFedH at varying PCL rounds

Round	MSE	MAE	MAPE (%)
$R_{pcl} = 1$	2.562±1.730	1.122±0.385	1.437±0.552
$R_{pcl} = 5$	2.548±1.745	1.119±0.389	1.433±0.560
$R_{pcl} = 10$	2.549±1.736	1.117±0.383	1.431±0.549
$R_{pcl} = 50$	2.565±1.762	1.118±0.381	1.430±0.542
$R_{pcl} = 100$	2.549±1.740	1.116±0.381	1.428±0.544

MAE: mean absolute error; MAPE: mean absolute percentage error; MSE: mean squared error; PCL: prototypical contrastive learning

Fig. 3. We show the visualizations of the representations obtained by PCFedH's enhanced clustering and the original data, separately. The visualization indicates that the application of prototypical cluster learning significantly enhances the clarity of clustering.

In summary, employing prototypical cluster learning enhances cluster separability and stability, surpassing traditional methods such as FedSoft clustering.

5.2.5 Effectiveness of personalization improvement

As shown in Fig. 4, we present the predictive outcomes for client 3 as a case study, highlighting the differential performance of PCFedH and FedProx. Client 3's dataset encompasses four sequential driving segments, characterized by HRs that surpass the average of the general population. PCFedH consistently delivers predictions that more accurately match the ground truth, particularly due to its capability to generate higher predictive values that accommodate the client's elevated HR level. Conversely, FedProx's predictions fall short, gravitating towards the mean levels of the broader population due to its lack of personalized training, thereby yielding less precise predictive values.

Moreover, differential performance between PCFedH and FedProx across all clients is detailed in the supplementary materials.

In summary, PCFedH consistently improves performance across clients in the Distracted dataset compared to FedProx. Visualizations of predictive

outcomes for client 3 highlight PCFedH's superior accuracy in reflecting individual characteristics, especially in scenarios with elevated HRs.

6 Conclusions

In our work, we contribute PCFedH, a novel FL framework designed for HR prediction. Addressing critical privacy concerns related to the HR data collection, PCFedH efficiently harnesses HR heterogeneity to provide tailored predictions.

Different from classical clustered FL, by integrating prototypical contrastive learning with clustered FL and applying the contextual consistency data augmentation technology, our method accurately classifies user data into distinct categories. This enables precise and personalized HR predictions while ensuring data privacy, even in non-IID scenarios with statistical heterogeneity among users. Simultaneously, prototype learning is used to aggregate prototypes among users to ease local data bias.

Through rigorous experimentation on two real-world datasets, our approach demonstrates significant improvements over state-of-the-art methods, achieving an average reduction of 3.1% in MSE compared with the second-best results across both datasets. Furthermore, our comprehensive experiments empirically validate the effectiveness of the key components of PCFedH.

Our work has two main limitations. First, the prototypical contrastive learning-based federated clustering module requires a predefined number of clustering types (K), which may need dynamic

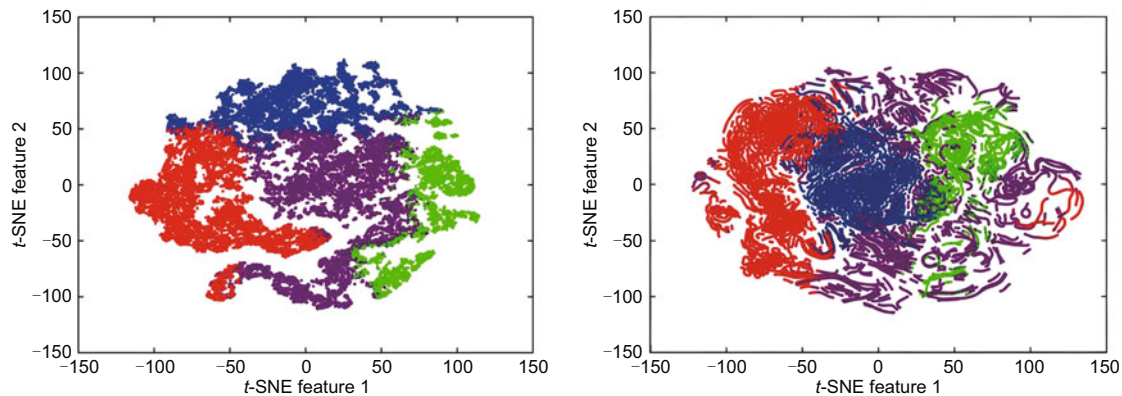


Fig. 3 Visualizations of PPG-DaLiA dataset representations. The left figure demonstrates the enhanced representation by PCFedH while the right one shows the original representation. Points of different colors represent different classes. The visualizations highlight how prototypical cluster learning remarkably improves clustering clarity. References to color refer to the online version of this figure

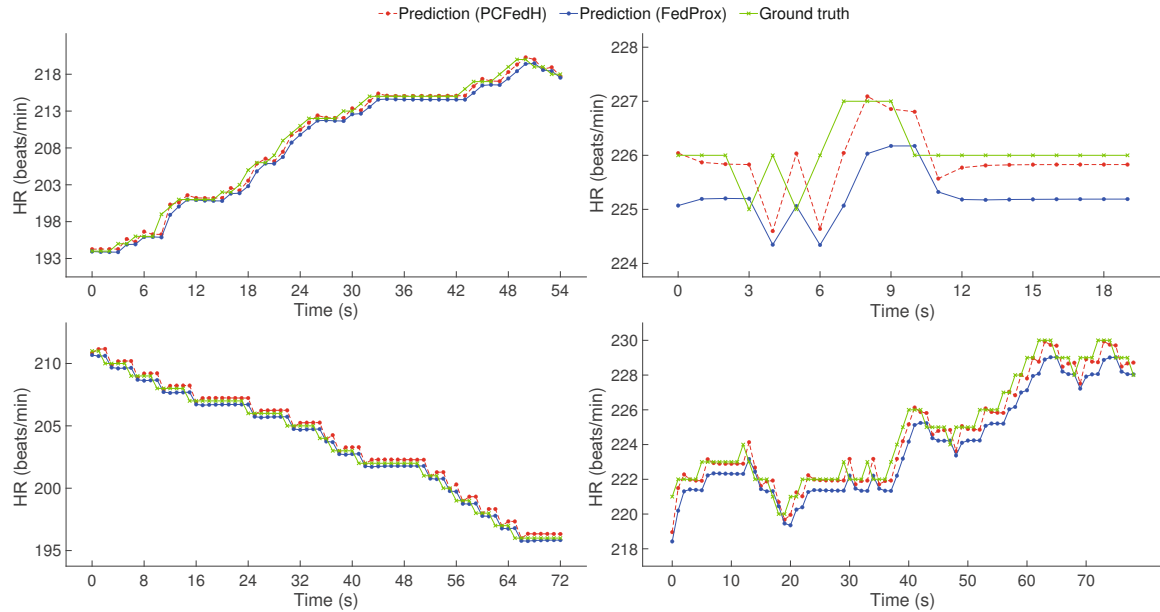


Fig. 4 Comparative analysis of heart rate (HR) predictions during consecutive driving segments. This set of four sub-graphs depicts the predictive performance of two methods, PCFedH and FedProx, against the ground truth for client 3's HR over time. The PCFedH predictions are consistently closer to the actual HR values, particularly in higher ranges, suggesting a better personalized fit for this client's data compared to the more generalized predictions of FedProx

adjustment as the diversity of activity conditions expands in the FL process. This could potentially affect the adaptability of our model. Second, our current experiments are limited to single-step predictions.

Improving the ability to predict future HR trends could provide a more comprehensive safeguard for clients' health.

For future work, we plan to enhance the client selection mechanism in FL, aiming to involve fewer clients in each training round to reduce costs effectively. Additionally, we aim to develop methods for dynamically adjusting the predefined cluster numbers (K) in real time to maintain model adaptability during federated clustering. Furthermore, we intend to extend our prediction capabilities beyond single-step forecasting to anticipate future HR trends more effectively, thereby enhancing health monitoring in FL settings.

Contributors

Yongjie YIN, Hui RUAN, Yang CHEN, Jiong CHEN, and Qingyuan GONG proposed the motivation of the study. Ziyue LI designed the PCFedH framework. Yongjie YIN and Hui RUAN performed the experiments and drafted the paper. Yipeng ZHOU and Xiang SU guided the convergence proof

of PCFedH. All the authors commented on previous versions of the paper, and read and approved the final version of the paper.

Conflict of interest

All the authors declare that they have no conflict of interest.

Data availability

The data that support the findings of this study are openly available in GitHub at <https://github.com/pcfhdh/pcfhdh>.

References

- Alharbi A, Alosaimi W, Sahal R, et al., 2021. Real-time system prediction for heart rate using deep learning and stream processing platforms. *Complexity*, 2021:5535734. <https://doi.org/10.1155/2021/5535734>
- Brisimi TS, Chen RD, Mela T, et al., 2018. Federated learning of predictive models from federated electronic health records. *Int J Med Inform*, 112:59-67. <https://doi.org/10.1016/j.ijmedinf.2018.01.007>
- Cai LX, Chen NY, Cao YZH, et al., 2023. FedCE: personalized federated learning method based on clustering ensembles. *Proc 31st ACM Int Conf on Multimedia*, p.1625-1633. <https://doi.org/10.1145/3581783.3612217>
- Chen J, Jiang YN, Huang ZX, et al., 2021. Fine-grained detection of driver distraction based on neural architecture

- search. *IEEE Trans Intell Transp Syst*, 22(9):5783-5801. <https://doi.org/10.1109/TITS.2021.3055545>
- Fang L, Liu XL, Su X, et al., 2020. Bayesian inference federated learning for heart rate prediction. Proc 9th EAI Int Conf on Wireless Mobile Communication and Healthcare, p.116-130. https://doi.org/10.1007/978-3-030-70569-5_8
- Ghosh A, Chung J, Yin D, et al., 2022. An efficient framework for clustered federated learning. *IEEE Trans Inform Theory*, 68(12):8076-8091. <https://doi.org/10.1109/TIT.2022.3192506>
- Jain AK, Dubes RC, 1988. Algorithms for Clustering Data. Prentice-Hall, Inc., Saddle River, USA.
- Kingma DP, Ba LJ, 2015. Adam: a method for stochastic optimization. 3rd Int Conf on Learning Representations, p.1-15.
- Li JN, Zhou P, Xiong CM, et al., 2021. Prototypical contrastive learning of unsupervised representations. <https://arxiv.org/abs/2005.04966>
- Li T, Sahu AK, Zaheer M, et al., 2020. Federated optimization in heterogeneous networks. *Proc Mach Learn Syst*, 2:429-450.
- McMahan B, Moore E, Ramage D, et al., 2017. Communication-efficient learning of deep networks from decentralized data. Proc 20th Int Conf on Artificial Intelligence and Statistics, p.1273-1282.
- Mehrang S, Pietila J, Tolonen J, et al., 2017. Human activity recognition using a single optical heart rate monitoring wristband equipped with triaxial accelerometer. European Medical and Biological Engineering Conf, p.587-590. https://doi.org/10.1007/978-981-10-5122-7_147
- Mu XT, Shen YL, Cheng K, et al., 2023. FedProc: prototypical contrastive federated learning on non-IID data. *Fut Gener Comp Syst*, 143:93-104. <https://doi.org/10.1016/j.future.2023.01.019>
- Oyeleye M, Chen TH, Titarenko S, et al., 2022. A predictive analysis of heart rates using machine learning techniques. *Int J Environ Res Public Health*, 19(4):2417. <https://doi.org/10.3390/ijerph19042417>
- Panwar M, Gautam A, Biswas D, et al., 2020. PP-net: a deep learning framework for PPG-based blood pressure and heart rate estimation. *IEEE Sensors J*, 20(17):10000-10011. <https://doi.org/10.1109/JSEN.2020.2990864>
- Patel M, Lal SKL, Kavanagh D, et al., 2011. Applying neural network analysis on heart rate variability data to assess driver fatigue. *Expert Syst Appl*, 38(6):7235-7242. <https://doi.org/10.1016/j.eswa.2010.12.028>
- Patidar S, Pachori RB, Acharya UR, 2015. Automated diagnosis of coronary artery disease using tunable-Q wavelet transform applied on heart rate signals. *Knowl-Based Syst*, 82:1-10. <https://doi.org/10.1016/j.knosys.2015.02.011>
- Qi Z, Meng L, Chen ZT, et al., 2023. Cross-silo prototypical calibration for federated learning with non-IID data. Proc 31st ACM Int Conf on Multimedia, p.3099-3107. <https://doi.org/10.1145/3581783.3612481>
- Rather AM, Agarwal A, Sastry VN, 2015. Recurrent neural network and a hybrid model for prediction of stock returns. *Expert Syst Appl*, 42(6):3234-3241. <https://doi.org/10.1016/j.eswa.2014.12.003>
- Reiss A, Indlekofer I, Schmidt P, et al., 2019. Deep PPG: large-scale heart rate estimation with convolutional neural networks. *Sensors*, 19(14):3079. <https://doi.org/10.3390/s19143079>
- Ruan YC, Joe-Wong C, 2022. FedSoft: soft clustered federated learning with proximal local updating. Proc 36th AAAI Conf on Artificial Intelligence, p.8124-8131. <https://doi.org/10.1609/aaai.v36i7.20785>
- Sattler F, Müller KR, Samek W, 2021. Clustered federated learning: model-agnostic distributed multitask optimization under privacy constraints. *IEEE Trans Neur Netw Learn Syst*, 32(8):3710-3722. <https://doi.org/10.1109/TNNLS.2020.3015958>
- Staffini A, Svensson T, Chung UI, et al., 2022. Heart rate modeling and prediction using autoregressive models and deep learning. *Sensors*, 22(1):34. <https://doi.org/10.3390/s22010034>
- Taamneh S, Tsiamyrtzis P, Dcosta M, et al., 2017. A multimodal dataset for various forms of distracted driving. *Sci Data*, 4(1):170110. <https://doi.org/10.1038/sdata.2017.110>
- Tan AZ, Yu H, Cui LZ, et al., 2023. Towards personalized federated learning. *IEEE Trans Neur Netw Learn Syst*, 34(12):9587-9603. <https://doi.org/10.1109/TNNLS.2022.3160699>
- Tan Y, Long GD, Ma J, et al., 2022a. Federated learning from pre-trained models: a contrastive learning approach. Proc 36th Conf on Neural Information Processing System, p.19332-19344.
- Tan Y, Long GD, Liu L, et al., 2022b. FedProto: federated prototype learning across heterogeneous clients. Proc 36th AAAI Conf on Artificial Intelligence, p.8432-8440. <https://doi.org/10.1609/aaai.v36i8.20819>
- van der Maaten L, Hinton G, 2008. Visualizing data using t-SNE. *J Mach Learn Res*, 9(86):2579-2605.
- Xu MA, Moreno A, Nagesh S, et al., 2022. PulseImpute: a novel benchmark task for pulsatile physiological signal imputation. Proc 36th Conf on Neural Information Processing Systems, p.26874-26888.
- Yao HX, Tang XF, Wei H, et al., 2019. Revisiting spatial-temporal similarity: a deep learning framework for traffic prediction. Proc 33rd AAAI Conf on Artificial Intelligence, p.5668-5675. <https://doi.org/10.1609/aaai.v33i01.33015668>
- Yaqoob MM, Nazir M, Khan MA, et al., 2023. Hybrid classifier-based federated learning in health service providers for cardiovascular disease prediction. *Appl Sci*, 13(3):1911. <https://doi.org/10.3390/app13031911>
- Yu ZT, Shen YM, Shi JG, et al., 2022. PhysFormer: facial video-based physiological measurement with temporal difference transformer. Proc IEEE/CVF Conf on Computer Vision and Pattern Recognition, p.4176-4186. <https://doi.org/10.1109/CVPR52688.2022.00415>
- Yue ZH, Wang YJ, Duan JY, et al., 2022. TS2Vec: towards universal representation of time series. Proc AAAI Conf on Artificial Intelligence, p.8980-8987. <https://doi.org/10.1609/aaai.v36i8.20881>
- Zhang J, Hua Y, Wang H, et al., 2023. FedALA: adaptive local aggregation for personalized federated learning. Proc AAAI Conf on Artificial Intelligence, p.11237-11244.
- Zhu HY, Xu JJ, Liu SQ, et al., 2021. Federated learning on non-IID data: a survey. *Neurocomputing*, 465:371-390. <https://doi.org/10.1016/j.neucom.2021.07.098>

List of supplementary materials

- 1 Personalized improvement among clients
- 2 Contrastive learning
- 3 Robustness evaluation of PCFedH
- 4 Convergence analysis

Table S1 Performance comparison of PCFedH and baseline methods in stimulus and normal conditions

Fig. S1 Performance comparison of FedProx and PCFedH in HR prediction