SSRG International Journal of Electronics and Communication Engineering
ISSN: 2348-8549/ https://doi.org/10.14445/23488549/IJECE-V12I9P101

Volume 12 Issue 9, 1-10, September 2025

Original Article

Federated Learning with Interpretable Deep Models for Diabetes Prediction in Non-IID Settings Using the Flower Framework

Prachi Gawande[1], Yogita Dubey[2], Punit Fulzele[3]

[1]*Department of Electronics Engineering, Yeshwantrao Chavan College of Engineering, Maharashtra, India.*
[2]*Department of Electronics & Telecommunication Engineering, Yeshwantrao Chavan College of Engineering, Maharashtra, India.*
[3]*SPDC Directorate of Research & Innovation Organization, Datta Meghe Institute of Higher Education & Research, Wardha, India.*

[1]*Corresponding Author : gawandeprachi7@gmail.com*

Received: 01 July 2025 Revised: 03 August 2025 Accepted: 02 September 2025 Published: 29 September 2025

Abstract - *In the era of privacy-preserving Machine Learning (ML), Federated Learning (FL) presents a transformative example for collaborative model training across distributed data sources without exposing sensitive information. This paper investigates the application of FL in healthcare using the Pima Indians Diabetes dataset, with a strong emphasis on non-Independent and Identically Distributed (non-IID) data partitioning, local client updates, and model interpretability. Three fully connected layers in a neural network and ReLU activations, implemented in PyTorch, are trained across five simulated clients using the Flower (FLWR) framework. The dataset is standardized, and clients receive shards of label-sorted data to replicate real-world heterogeneity across healthcare providers. Each Client trains its model using the Adam optimizer and cross-entropy loss, with local training loss monitored over multiple epochs. Post-training, interpretability techniques-LIME (Local Interpretable Model-agnostic Explanations)- were employed to explain distinct predictions and global feature influence. Experimental results demonstrate that while federated learning can achieve reasonable performance in non-IID settings, interpretability insights vary significantly across clients due to data distribution disparities. The findings highlight the need for client-aware personalization and future enhancements in federated optimization strategies, communication efficiency, and explainable AI in sensitive domains like healthcare.*

Keywords - *Federated learning, Privacy-preserving machine learning, Flower framework, Pytorch, Pima Indians diabetes, Neural networks, Healthcare AI, Data privacy, Distributed learning, Model aggregation.*

1. Introduction

The rapid digitization of healthcare systems worldwide has led to an unprecedented accumulation of Electronic Health Records (EHRs), which encapsulate valuable clinical, demographic, and biometric information. These datasets possess immense potential for powering predictive modelling applications aimed at early diagnosis, personalized treatment planning, and clinical decision support systems. However, leveraging this data at scale poses significant challenges due to strict regulatory frameworks and ethical constraints surrounding patient privacy. Legal mandates like the Health Insurance Portability and Accountability Act (HIPAA) in the United States [1] and the General Data Protection Regulation (GDPR) in the European Union [2] prohibit the unrestricted sharing of sensitive medical data across institutional or geographic boundaries. As a result, traditional centralized machine learning paradigms that require raw data aggregation are often infeasible in healthcare settings [3]. This critical bottleneck has catalysed the exploration of privacy-preserving machine learning paradigms, among which Federated Learning (FL) has appeared as a particularly promising approach [4].

FL enables various decentralized clients-such as hospitals, clinics, or research centers-to collaboratively train a shared global model without transmitting any raw data to a central server. Instead, each Client computes updates (e.g., gradients or model weights) on its local data and communicates only these updates to a coordinating server, where they are aggregated (e.g., via Federated Averaging) [5]. This decentralized learning framework aligns well with the privacy-sensitive nature of healthcare data and facilitates knowledge sharing across institutions without violating legal or ethical constraints.

Moreover, FL accommodates the statistical heterogeneity that is characteristic of real-world healthcare datasets. Unlike conventional machine learning scenarios that assume Identically and Independently Distributed (IID) data, patient records are inherently non-IID across healthcare providers due to differences in demographics, medical practices, equipment, and regional disease prevalence [6]. These disparities introduce significant challenges for federated optimization algorithms, particularly in terms of model convergence, fairness, and generalization [7]. Yet, if addressed properly, such heterogeneity can also serve as a source of richness and diversity in model training, enhancing the robustness of predictive systems [8].

This paper investigates the application of federated learning to diabetes prediction using the Pima Indians Diabetes dataset, with an explicit focus on modelling non-IID data distributions and improving model interpretability. The implementation leverages the Flower (FLWR) framework [9], an open-source platform for scalable and customizable federated learning experimentation. The federated environment was stimulated with five clients; each was assigned a label-skewed partition of the dataset to mimic heterogeneity encountered in clinical practice.

A fully connected neural network with three hidden layers is trained locally at each Client using the PyTorch deep learning framework. Training is conducted using the Adam optimizer and cross-entropy loss, with model updates aggregated centrally via FedAvg [5]. Recognizing that black-box models are insufficient for clinical deployment without transparency, a federated learning pipeline is complemented with post-hoc interpretability techniques. Specifically, the application of Local Interpretable Model-agnostic Explanations (LIME) [10] and SHapley Additive exPlanations (SHAP) [11] to elucidate both local and global aspects of the model's decision-making process. These tools enable us to examine how feature contributions differ across clients and provide actionable insights into the model's predictions.

The key contributions are as follows:

- Federated Implementation in Non-IID Context: Presented a practical, end-to-end federated learning pipeline using Flower and PyTorch, tailored to simulate realistic non-IID data distributions commonly encountered in healthcare scenarios.
- Empirical Analysis of Client-Specific Behavior: Visualizing and monitoring local training loss across clients provided an in-depth look at how data heterogeneity influences convergence dynamics and model behaviour.
- Integrating Interpretability into FL: The demonstration of how LIME and SHAP can be used to interpret federated models, revealing both shared and client-specific feature importances, thus bridging the gap between black-box modelling and clinical interpretability requirements.
- Discussion on Ethical and Technical Implications: Provided a critical analysis of the challenges and opportunities that arise at the intersection of federated learning, interpretability, and healthcare, including the implications for fairness, personalization, and regulatory compliance.

2. Literature Survey

The emergence of Federated Learning (FL) has unlocked new possibilities in machine learning where data privacy, decentralization, and secure computation are paramount-especially in subtle domains such as healthcare. Traditional machine learning models often rely on centralized training data, raising legal and ethical issues under regulations like GDPR and HIPAA. McMahan et al. [5] introduced the Federated Averaging (FedAvg) algorithm, allowing decentralized clients without sharing raw data to train a global model collaboratively. This innovation paved the way for a new generation of learning paradigms. However, FedAvg is sensitive to statistical heterogeneity in client data distributions-a concern amplified in clinical environments where demographic and regional variations abound. Following this, Kairouz et al. [7] provided a comprehensive roadmap of open problems in FL, highlighting the critical importance of handling non-IID data and the lack of theoretical convergence guarantees in such settings. Zhao et al. [8] systematically investigated the performance degradation that occurs when client datasets are not identically distributed, showing that even slight imbalances can drastically reduce global model accuracy.

In response to these foundational challenges, researchers have developed various strategies to ease the effects of non-IID data. Li et al. [7] explored the personalization of federated models to align better with local client distributions, introducing solutions such as meta-learning and clustering-based FL. Wang et al. [10] proposed a differentially private FL framework that balances client contribution using adaptive weight adjustments, thereby improving generalizability while preserving privacy. These approaches mark significant progress in making FL more robust and practical for real-world applications. However, most solutions still focus predominantly on global performance, with limited attention to how individual clients learn from their local data-a crucial consideration in medical contexts where local model reliability is vital.

The application of FL in healthcare has received considerable attention, especially after the COVID-19 pandemic, which emphasized the importance of data collaboration without compromising patient confidentiality. Sheller et al. [20] demonstrated a landmark implementation of FL for brain tumour segmentation using data from multiple institutions without data sharing. Their study validated the feasibility of collaborative learning in real clinical workflows.

Dayan et al. [21] extended this by training FL models across 20 hospitals for COVID-19 prognosis prediction, achieving state-of-the-art accuracy while respecting strict privacy protocols. Similarly, a patient-centric FL pipeline for privacy-preserving medical image diagnosis was proposed. Their model integrated resource-efficient techniques to reduce communication overhead, making it suitable for real-world deployment. These contributions showcase FL's ability to bridge data silos in healthcare, but they often involve complex medical imaging data or large-scale infrastructures, limiting generalizability to simpler clinical datasets like tabular records.

Despite these advances, one key limitation in FL research is the lack of model interpretability. Medical applications require transparency in decision-making, especially when AI models are used to support clinical decisions. Model-agnostic interpretability tools like LIME (Local Interpretable Model-agnostic Explanations) introduced by Ribeiro et al. [10], and SHAP (SHapley Additive exPlanations) by Lundberg et al. [11], have proven effective in explaining black-box models. However, most studies applying LIME and SHAP operate in centralized settings. Previously an author attempted to bridge this gap by applying SHAP in federated environments to explain global model decisions. While this was a step forward, it did not address how explanations might vary at the client level. An interpretable FL framework for medical text classification using attention-based mechanisms, showing that local attention weights could reflect linguistic features learned by each Client. Still, their study was domain-specific and lacked generalizability to numerical or tabular datasets like those commonly found in EHRs.

The need for client-level interpretability in federated settings has been emphasized in several recent works. An author investigated how non-IID training leads to divergent feature learning across clients, suggesting the necessity of localized explanation strategies.

They highlighted that even under identical architectures and hyperparameters, the decision boundaries formed by client models can differ significantly based on their unique training distributions. A comparative analysis of LIME explanations across federated client models and found that inconsistencies in feature attribution could reveal model drift or overfitting. Their findings support the argument that federated interpretability is essential for transparency, debugging, and trust calibration. A multi-modal federated architecture incorporating interpretable layers, enabling clinicians to visualize and validate the influence of text and image data on diagnosis predictions. While these studies advance the field of federated interpretability, most focus either on visual or unstructured data and do not directly analyze how tabular, structured health data is learned under federated settings.

In contrast to these approaches, the current work presents a lightweight, client-level interpretability pipeline using LIME on federated models trained under non-IID conditions using the Pima Indians Diabetes dataset. By assigning label-skewed data to clients and applying LIME explanations post-training, the study investigates how each model learns feature importance uniquely, depending on its local dataset characteristics. This approach offers a novel perspective on FL behavior under heterogeneity, providing a critical step toward explainable, privacy-preserving, and clinically relevant AI systems.

2. Methodology

In this study, designed and implemented a Federated Learning (FL) system was designed and implemented to evaluate the impact of non-independent and identically distributed (non-IID) data on model performance and interpretability in a healthcare prediction task.

The system simulates a collaborative learning environment among five clients using the Flower (FLWR) framework [12] and PyTorch [13], with a focus on diabetes prediction using the Pima Indians Diabetes dataset [14].

2.1. Data Acquisition and Preprocessing

The dataset was obtained from a public repository and contains 768 samples with 8 clinical features related to diabetes risk, laterally with a binary outcome label indicating the presence or absence of diabetes [16].

To simulate real-world clinical heterogeneity, non-IID conditions were introduced by sorting the data by class labels and then partitioning it into shards as shown in Figure 1. Each of the five clients received two label-skewed shards, resulting in imbalanced class distributions per Client [17].

Standard preprocessing steps were applied. First, features were homogenised to zero mean and unit variance using z-score normalization with StandardScaler [16] to improve training stability. The dataset was then converted to PyTorch tensors, and a validation set was created by randomly splitting 20% of the data using train_test_split from Scikit-learn [17]. This validation set remained centralized for post-training model interpretation.

The dataset features are standardized using z-score normalization, which helps stabilize and accelerate training as shown in Equation (1):

$$x' = \frac{x-\mu}{\sigma} \tag{1}$$

Where

x - original feature value,
μ - mean of the feature,
σ - standard deviation.

Fig. 1 Non-IID data partitioning for federated clients

This ensures all features have zero mean and unit variance, making gradient-based optimization more effective [18].

2.2. Flower (FLWR) Framework Architecture

In Federated Learning (FL), a central server communicates with multiple clients that are part of a collaborative network, commonly referred to as a federation.

The server's primary function is to manage and orchestrate the training process, while each Client is responsible for executing the assigned tasks and sending the outcomes back to the server.

This architecture is often described as a hub-and-spoke model, as illustrated in Figure 1.

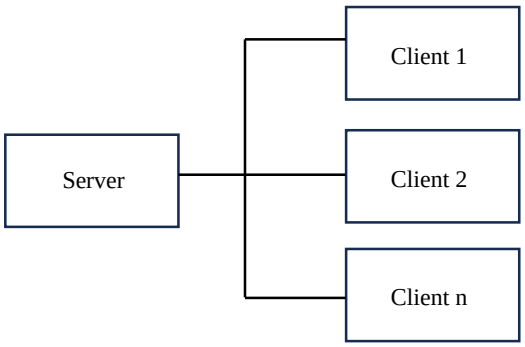

Fig. 2 Hub and Spoke topology in federated learning (one server, multiple clients)

In real-world deployments, running multiple projects within a single federation is common. Each project may utilize distinct hyperparameters, model architectures, aggregation strategies, or even different machine learning frameworks such as PyTorch or TensorFlow.

To accommodate this flexibility, Flower separates both the server and client components into two parts: one is persistent and handles network communication, while the other is temporary and runs task-specific logic. As illustrated in Figure 2, a Flower server is composed of the Super Link and the Server App.

Super Link is a persistent process that transmits task instructions to clients (referred to as Super Nodes) and collects the corresponding results.

Server App is a transient process containing project-specific logic that defines all server-side components of a federated learning system, including client selection, configuration, and result aggregation. This component is typically developed by AI researchers and engineers when creating Flower-based applications. As depicted in Figure 3, a Flower client comprises two components: Super Node and Client App.

Super Node is a long-running process that establishes a connection with the Super Link, requests a task, performs those tasks (such as training a model on local data), and returns the outcomes to Super Link. Client App is a short-lived component containing project-specific code that defines client-side operations such as local model training, evaluation, and any necessary pre- or post-processing. Like the Server App, this is implemented by AI researchers and engineers when building Flower applications.

Within the framework of federated learning, clients play a central role-they possess the training data and carry out the actual training processes. This is why Flower refers to them as Super Nodes, while the Super Link serves as the coordinating element that bridges all the Super Nodes together [9].

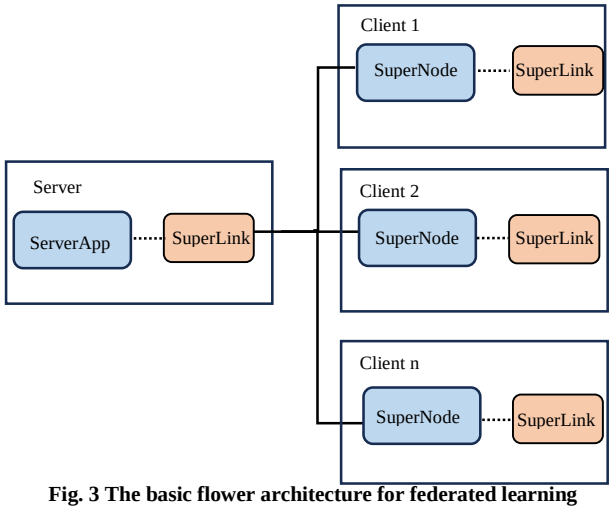

Fig. 3 The basic flower architecture for federated learning

2.3. Model Architecture

Implementation of a compact, fully connected feedforward neural network designed for tabular binary classification. The architecture consists of three linear layers:

The first hidden layer has 16 neurons with ReLU activation. The second layer contains 8 neurons with ReLU activation. The output layer contains 2 neurons corresponding to the binary classes, followed by a SoftMax function.

This architecture was selected for its expressiveness and computational efficiency balance, making it suitable for federated environments with limited compute resources [5].

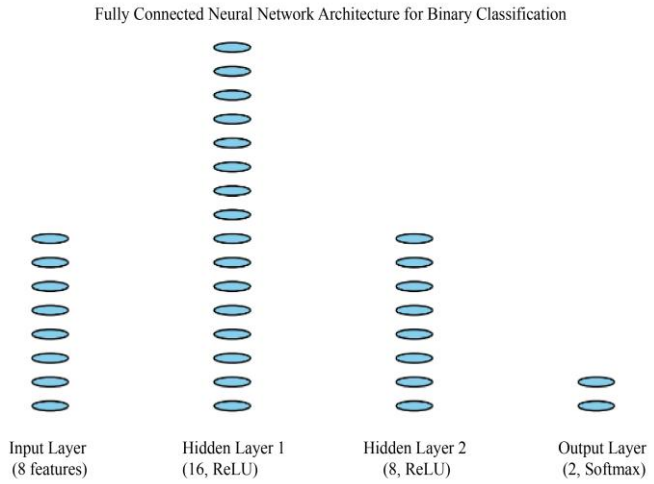

Fig. 4 Fully connected neural network architecture for binary classification

Let $x \in \mathbb{R}^8$ be the input feature vector. The model consists of the following layers, as shown in Equations (2), (3), and (4):

First Hidden Layer

$$h_1 = Re\,L\,U(W_1 x + b_1), W_1 \in \mathbb{R}^{16\times 8} \quad (2)$$

Second Hidden Layer

$$h_1 = Re\,L\,U(W_2 h_1 + b_2), W_2 \in \mathbb{R}^{8\times 16} \quad (3)$$

Output Layer

$$y = \text{Softmax}(W_3 h_2 + b_3), W_3 \in \mathbb{R}^{2\times 8} \quad (4)$$

The final output $y \in \mathbb{R}^2$represents the predicted probability distribution over the two classes (diabetic or non-diabetic).

2.4. Federated Learning Setup

The FL system was built using the Flower framework [12], which provides abstractions for simulating federated clients and server coordination. Each Client runs an instance of the Diabetes Client class, a subclass of fl. Client.NumPyClient, which encapsulates local training, evaluation, and parameter synchronization logic.

Each client model was independently initialized [18] and trained on its local dataset using the Adam optimizer with a learning rate of 0.01. The loss function used was categorical cross-entropy, suitable for multi-class classification problems [10]. During training, the model was updated over 5 local epochs per round. After local training, each Client sent its updated parameters to the server, which performed weighted aggregation using the standard Federated Averaging (FedAvg) strategy [19].

The simulation was initially performed manually for one Client to verify convergence and training dynamics. As shown in Figure 5, training loss per epoch was logged and plotted to visually assess optimization behavior in the non-IID setting.

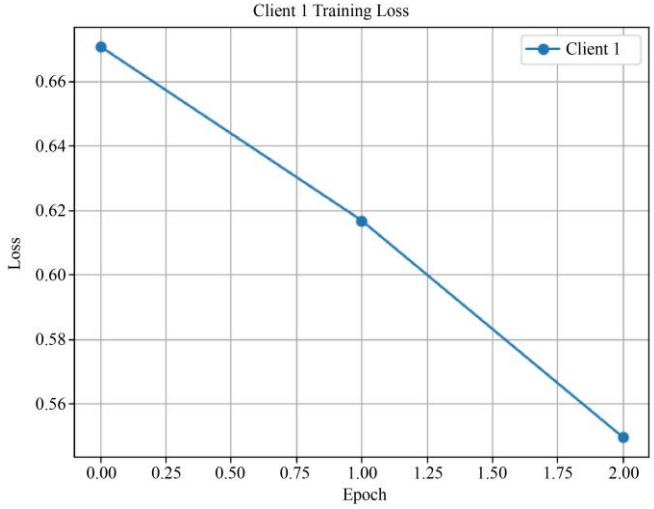

Fig. 5 Training loss per epoch

2.5. Model Interpretability

To provide transparency in model decision-making and highlight the effects of data heterogeneity on learned representations, we incorporated post-hoc interpretability techniques that explain how each federated Client's model arrives at its predictions under non-IID data distributions. We employed Local Interpretable Model-Agnostic Explanations (LIME) [20] as a post-hoc interpretability technique. LIME provides feature-level attribution by approximating the model's complex, black-box decision boundary with a simpler, locally interpretable surrogate model. Specifically, analyzed the same fixed sample from the centralized validation dataset across all five client models to examine how differences in local training data affect the individual Client's decision-making processes.

Mathematically, LIME seeks to learn an explanation model selected from a class of interpretable models G, by minimizing the following objective function as shown in Equation (5).

$$\hat{g} = argmin\ L(f, g, \pi_x) + \Omega(g) \quad (5)$$

Where,

$f$is the original complex model (e.g., neural network) trained on each Client.

$g$is the interpretable surrogate model, such as a sparse linear regressor.

$L(f, g, \pi_x)$ is a local fidelity loss function that measures how well $g$approximates $f$in th vicinity of the instance x, weighted by the locality kernel π_x.

$\pi_x(z)$ defines the proximity measure, often modeled using an exponential kernel:

$$\pi_x = \pi r^2 \quad (5)$$

Where is a distance metric (e.g., Euclidean or cosine distance) between the instance of interest and a perturbed sample z, and is a kernel width parameter controlling locality.

$\Omega(g)$ is a complexity penalty term that ensures it remains interpretable by enforcing sparsity or simplicity.

By approximating the behaviour of each federated model locally around a specific input instance, LIME reveals the most influential features contributing to each prediction. This approach allows us to contrast how individual clients, trained on skewed local datasets, weigh feature importance differently. The resulting explanations are instrumental in interpreting client-specific model behaviour and assessing the impact of data heterogeneity on model logic.

In the results section, the LIME-generated explanations were presented and analysed for each of the five clients, offering insights into how local data distributions shape the decision boundaries of federated models.

3. Results and Discussion

For Client 1, Figure 6 shows that the most influential feature was Pregnancies > 0.64, which contributed positively to the diabetes prediction with a weight of approximately +0.30. This was followed closely by Diabetes Pedigree Function > 0.42, contributing around +0.29, and BMI > 0.57 at approximately +0.18. These values indicate that the model strongly associated these three features with an increased likelihood of diabetes, making them the primary drivers of its decision. Additionally, a moderate positive contribution came from 0.07 < Glucose <= 0.79 with a weight of +0.08.

The only negative contributor was Insulin <= -0.69, which exerted a small suppressing influence of −0.04 on the diabetic classification. The remaining features-Age, Blood Pressure, and Skin Thickness-had negligible weights near zero, reflecting minimal involvement in the prediction. This behaviour suggests that the Client 1 model was trained on data where hereditary and lifestyle factors strongly aligned with diabetes diagnoses, guiding its classification logic.

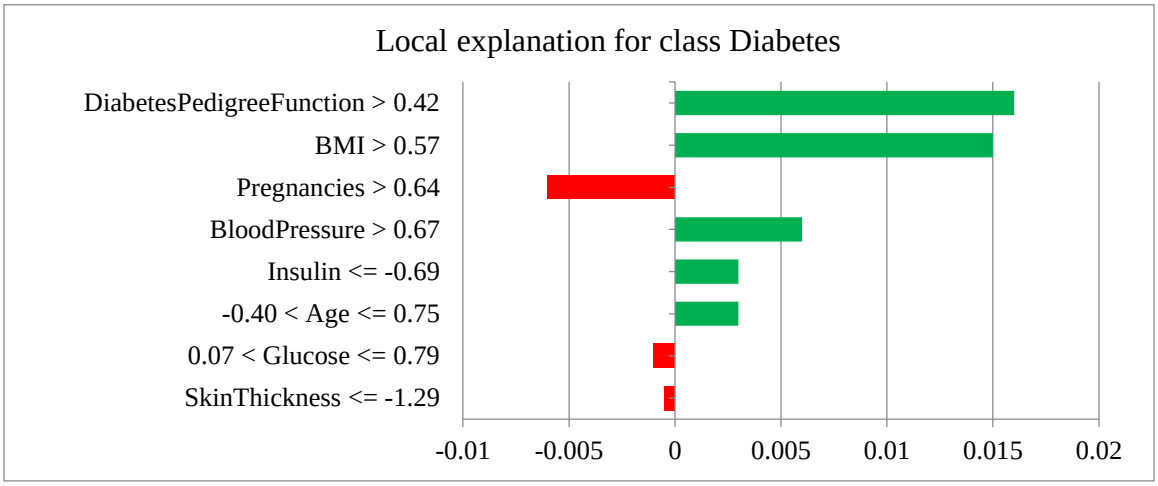

Fig. 6 Client 1 Feature impact on diabetes classification

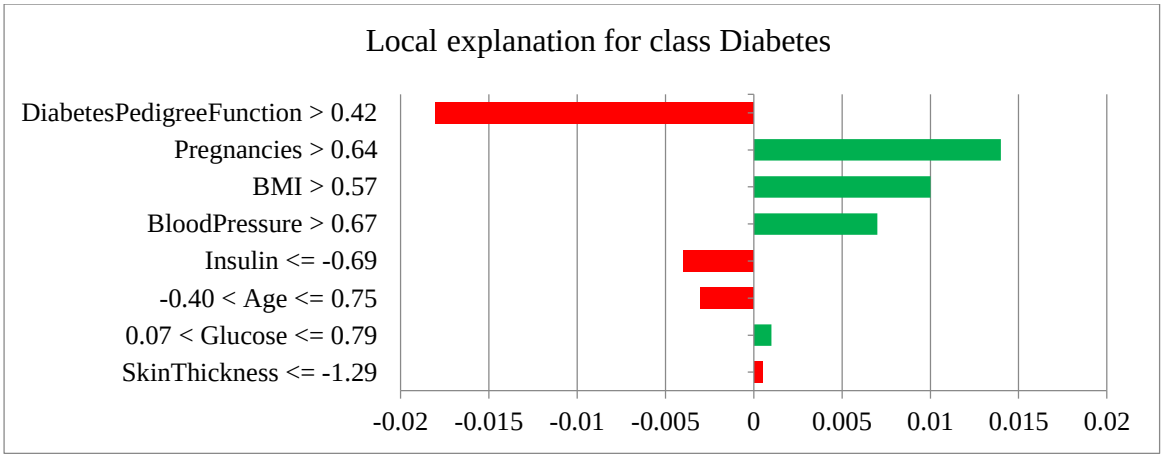

Fig. 7 Client 2 Feature impact on diabetes classification

In contrast to Client 1, the model for Client exhibited a very different attribution profile. As shown in Figure 7, the top contributor was again Pregnancies > 0.64, but its influence was more moderate at approximately +0.012. BMI > 0.57 and Blood Pressure > 0.67 followed with contributions of around +0.010 and +0.008, respectively. However, a surprising deviation emerged: Diabetes Pedigree Function > 0.42 contributed negatively with a weight of −0.018, indicating that this feature reduced the probability of predicting diabetes for this specific instance-opposite to the pattern seen in Client 1. Minor negative weights were also seen for Insulin <= −0.69 (−0.004) and Age (−0.003)

Glucose and Skin Thickness had almost no influence. These attributions imply that Client 2's training data may have involved a nonstandard relationship between genetic markers and diabetes presence, prompting the model to downplay typical clinical risk indicators like pedigree.

The LIME output for Client 3 reflected a more ambivalent model, as indicated by the relatively small contribution magnitudes across the board. As shown in Figure 8, the most positively weighted feature was Diabetes Pedigree Function > 0.42, which contributed about +0.009, followed by Blood Pressure > 0.67 and BMI > 0.57, with contributions of +0.007 and +0.006, respectively. Interestingly, Pregnancies > 0.64 and Insulin <= −0.69 both contributed negatively, with weights of approximately −0.005 each. The remaining features had minimal impact, clustered around ±0.001.

The low attribution values across all dimensions indicate that this model lacked dominant decision drivers, likely due to either a noisy local dataset or weak class separation in the training distribution. This could lead to more conservative or uncertain classifications compared to other clients.

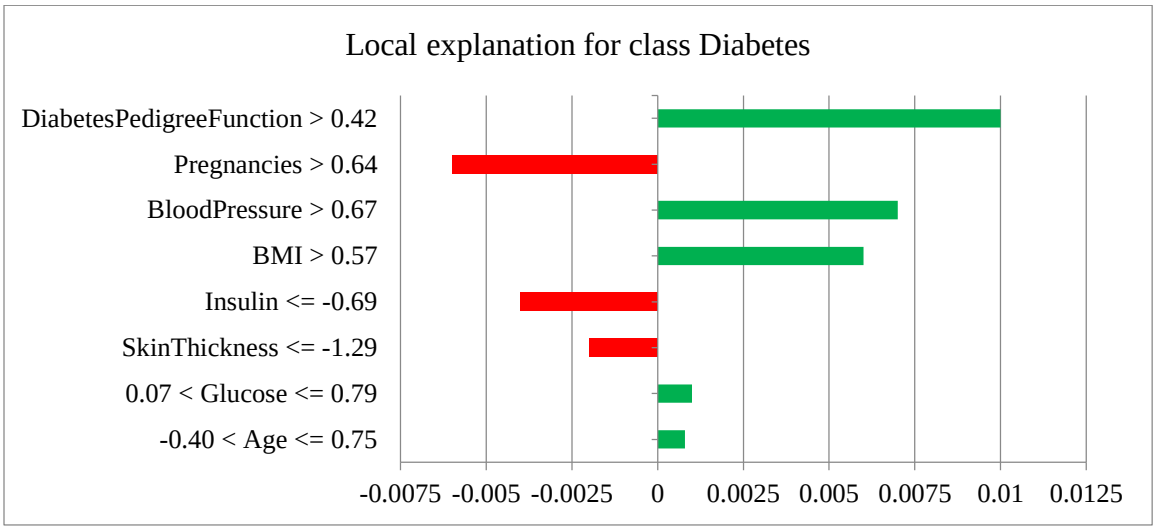

Fig. 8 Client 3 Feature impact on diabetes classification

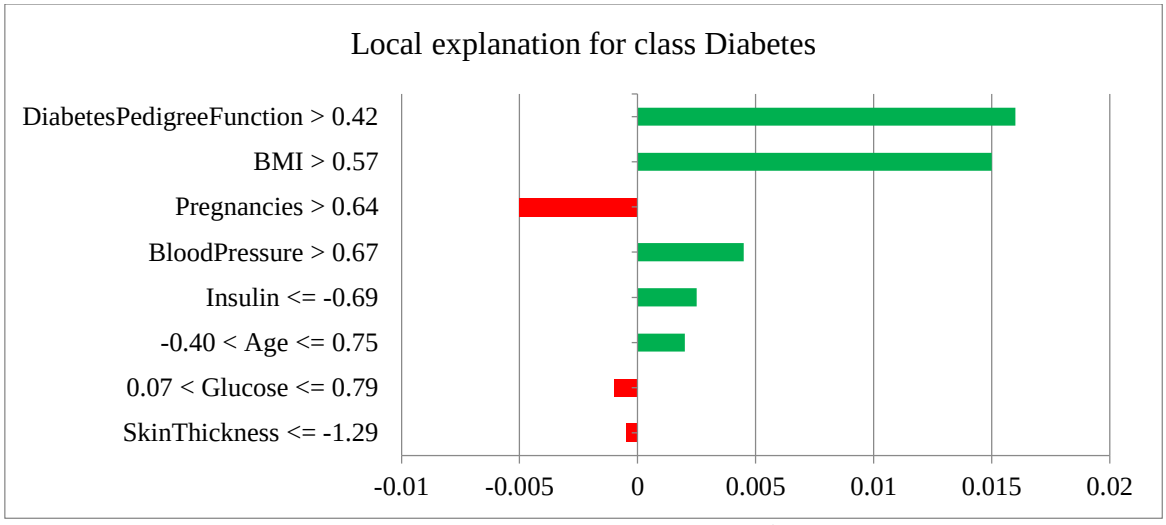

Fig. 9 Client 4 Feature impact on diabetes classification

Client 4's explanation, shown in Figure 9, is a relatively confident model, with strong positive influence from Diabetes Pedigree Function > 0.42 and BMI > 0.57, contributing approximately +0.016 and +0.014, respectively. However, unlike Client 1, Pregnancies > 0.64 contributed negatively (−0.005), meaning this model viewed high pregnancy count as decreasing the likelihood of diabetes for this instance. This highlights a reversal in learned feature associations, likely caused by local population skew (e.g., a younger or healthier cohort with high pregnancy counts). Other positive contributors included Blood Pressure > 0.67 (+0.006), Insulin <= −0.69 (+0.004), and Age (+0.003), while Glucose and Skin Thickness again showed minor weights.

This configuration indicates a model with stronger reliance on biological and metabolic markers, though with altered demographic interpretation relative to other clients. The most striking divergence occurred in Client 5's explanation. Pregnancies > 0.64 had the strongest positive effect, contributing around +0.033, followed by Diabetes Pedigree Function > 0.42 at +0.018. However, several classic risk features showed negative contributions: BMI > 0.57 (−0.011), Insulin <= −0.69 (−0.010), and Skin Thickness <= −1.29 (−0.009). These values suggest that, for this Client's model, elevated BMI and insulin resistance were negatively correlated with diabetes prediction-a stark contrast to established clinical patterns and previous clients, as shown in Figure 10.

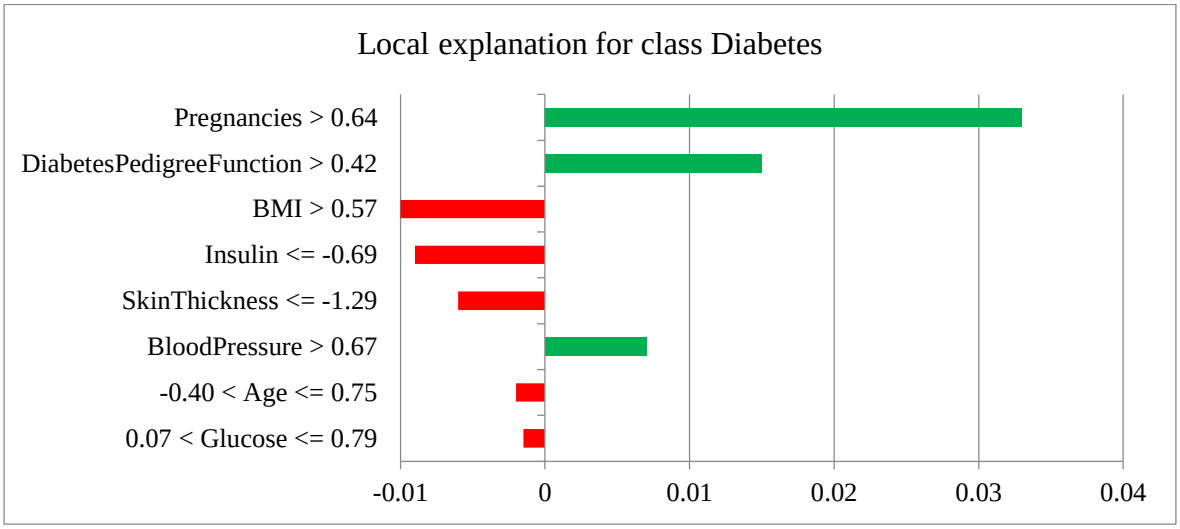

Fig. 10 Client 5 Feature impact on diabetes classification

The remaining features showed small influences: Blood Pressure was positive (+0.006), while Age and Glucose were weakly negative. This inverse attribution may result from a local dataset containing non-diabetic individuals with high BMI or Insulin, thus leading the model to internalize misleading associations.

Table 1. LIME explanations

Feature	Client 1	Client 2	Client 3	Client 4	Client 5
Pregnancies	+ Strong	- Weak	-	-	+ Strong
Diabetes Pedigree Function	+ Strong	- Strong	+	+ Strong	+ Moderate
BMI	+	+	+	+ Strong	-
Insulin	-	-	-	+	-
Blood Pressure	~	+	+	+	+
Age	Minimal	Minimal	Minimal	+	-
Skin Thickness	Negligible	Minimal	-	-	-

The LIME explanations in Table 1 reveal distinct patterns of feature importance across the five federated clients, highlighting how non-IID training data shape each model's decision logic. For instance, Pregnancies emerge as a strong positive predictor in Clients 1 and 5, but flip to a negative influence in Clients 3 and 4, suggesting that the label-skewed partitions at those sites associate higher pregnancy counts with lower diabetes risk. The Diabetes Pedigree Function consistently contributes positively-most strongly in Clients 1, 4, and moderately in Client 5-but is negatively weighted in Client 2, indicating local data differences that reverse its typical risk signal. BMI is uniformly positive except at Client 5, where it becomes a weak negative, again underscoring local heterogeneity.

Metabolic markers like Insulin are negative predictors in four out of five clients, yet Client 4 exhibits a strong positive weight, perhaps reflecting a different distribution of high-insulin cases at that site. Blood Pressure shows minimal or mixed effects in Client 1, but is positively associated with diabetes risk in all other clients.

Demographic factors also vary: Age has only a negligible influence on Clients 1–3, turns positive at Client 4, and is slightly negative at Client 5. Finally, Skin Thickness is largely uninformative-negligible or minimal in Clients 1 and 2 and negatively weighted elsewhere-indicating that this feature carries little consistent signal across the heterogeneous datasets.

Together, these client-by-client contrasts emphasize the necessity of interpretability in federated learning, as identical architectures can learn qualitatively different decision rules when trained on non-IID data.

4. Conclusion

In this study, a federated learning pipeline for diabetes prediction using the Pima Indians Diabetes dataset under non-IID conditions, leveraging the Flower framework and PyTorch. Through label-skewed data partitioning across five simulated clients, the experiments demonstrated that while federated averaging can produce a reasonably accurate global model, individual client models exhibit substantial variability in convergence behavior and decision logic. Post-hoc interpretability via LIME revealed pronounced differences in feature attributions-such as the reversal of Pregnancy and BMI weights in certain clients-underscoring that identical network architectures can internalize qualitatively different patterns when trained on heterogeneous data. These findings highlight two key implications: (1) interpretability is indispensable in federated healthcare applications to surface client-specific biases and ensure trust, and (2) statistical heterogeneity must be explicitly addressed to achieve both equitable performance and consistent model behavior across sites.

References

[1] Health Insurance Portability and Accountability Act of 1996 (HIPAA), U.S. Department of Health & Human Services, 2024. [Online]. Available: https://www.cdc.gov/phlp/php/resources/health-insurance-portability-and-accountability-act-of-1996-hipaa.html

[2] General Data Protection Regulation (GDPR), Intersoft Consulting. [Online]. Available: https://gdpr-info.eu/

[3] Alvin Rajkomar, Jeffrey Dean, and Isaac Kohane, "Machine Learning in Medicine," *New England Journal of Medicine*, vol. 380, no. 14, pp. 1347-1358, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[4] Qiang Yang et al., "Federated Machine Learning: Concept and Applications," *ACM Transactions on Intelligent Systems and Technology*, vol. 10, no. 2, pp. 1-19, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[5] Brendan McMahan et al., "Communication-Efficient Learning of Deep Networks from Decentralized Data," *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, vol. 54, pp. 1-11, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[6] Tian Li et al., "Federated Learning: Challenges, Methods, and Future Directions," *IEEE Signal Processing Magazine*, vol. 37, no. 3, pp. 50-60, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[7] M. Kairouz et al., "Advances and Open Problems in Federated Learning," *Foundations and Trends® in Machine Learning*, vol. 14, no. 1-2, pp. 1-210, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[8] Yue Zhao et al., "Federated Learning with Non-IID Data," *arXiv Preprint*, pp. 1-12, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[9] Daniel J. Beutel et al., "Flower: A Friendly Federated Learning Research Framework," *arXiv Preprint*, pp. 1-15, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[10] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin, "Why Should I Trust You?": Explaining the Predictions of Any Classifier," *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1135-1144, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[11] Scott M. Lundberg, and Su-In Lee, "A Unified Approach to Interpreting Model Predictions," *Advances in Neural Information Processing Systems*, vol. 30, pp. 1-10, 2017. [Google Scholar] [Publisher Link]

[12] Adam Paszke et al., "PyTorch: An Imperative Style, High-Performance Deep Learning Library," *Advances in Neural Information Processing Systems* vol. 32, pp. 8024-8035, 2019. [Google Scholar] [Publisher Link]

[13] Pima Indians Diabetes Database, National Institute of Diabetes and Digestive and Kidney Diseases, Kaggle. [Online]. Available: https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database

[14] Isabelle Guyon et al., *Feature Extraction: Foundations and Applications*, Springer Berlin Heidelberg, pp. 1-778, 2008. [Google Scholar] [Publisher Link]

[15] Fabian Pedregosa et al., "Scikit-Learn: Machine Learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825-2830, 2011. [Google Scholar] [Publisher Link]

[16] Diederik P. Kingma, and Jimmy Lei Ba, "Adam: A Method for Stochastic Optimization," *arXiv Preprint*, pp. 1-15, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[17] Yu Cheng et al., "A Survey on Model Compression and Acceleration for Deep Neural Networks," *arXiv Preprint*, vol. 53, no. 4, pp. 2649-2677, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[18] Ian Goodfellow, Yoshua Bengio, and Aaron Courville, *Deep Learning*, MIT Press, pp. 1-775, 2016. [Google Scholar] [Publisher Link]

[19] Kang Wei et al., "Federated Learning with Differential Privacy: Algorithms and Performance Analysis," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 3454-3469, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[20] Micah J. Sheller et al., "Federated Learning in Medicine: Facilitating Multi-Institutional Collaborations without Sharing Patient Data," *Scientific Reports*, vol. 10, pp. 1-12, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[21] Ittai Dayan et al., "Federated Learning for Predicting Clinical Outcomes in Patients with COVID-19," *Nature Medicine*, vol. 27, no. 10, pp. 1735-1743, 2021. [CrossRef] [Google Scholar] [Publisher Link]