OPEN ACCESS

Int. Jr. of Contemp. Res. in Multi. *PEER-REVIEWED JOURNAL* *Volume 5 Issue 1 [Jan- Feb] Year 2026*

Research Article

Federated Proximal in Privacy-Preserving for Disease Prediction Using Heterogeneous Healthcare Data

Amit Walia [1*], Dr. Ravinder Singh Madhan [2]

[1] Ph. D., Research Scholar, Department of Computer Science and Engineering, IEC University, Baddi, Solan, Himachal Pradesh, India

[2] Associate Professor, Department of Computer Science and Engineering, IEC University, Baddi, Solan, Himachal Pradesh, India

Corresponding Author: * Amit Walia

DOI: https://doi.org/10.5281/zenodo.21020355

Abstract

Privacy-preserving federated learning enables collaborative disease prediction while safeguarding sensitive patient data. This study explores the Federated Proximal (FedProx) algorithm within a federated learning framework to address the challenges of heterogeneous healthcare data. A simulated network of healthcare providers trains a shared model for disease prediction, particularly heart disease, using a synthetic multi-feature health dataset. Our methodology integrates data preprocessing techniques (handling missing values, mitigating outliers, normalisation, and anonymisation) and feature engineering (feature extraction, principal component analysis, and feature importance evaluation). FedProx-based training, coupled with split learning, enhances privacy and mitigates data heterogeneity. Experimental results demonstrate that the FedProx federated model achieves high accuracy, F1-score, and ROC-AUC, comparable to a centralised model, while ensuring strict privacy preservation. FedProx improves training stability across non-IID data sources, outperforming standard federated averaging (FedAvg). Feature importance analysis highlights Age, BMI, blood pressure, and sleep duration as key predictors and principal component analysis (PCA) confirms that two components capture 95% of data variance, validating dimensionality reduction techniques. This research confirms the viability of FedProx-enhanced federated learning for privacy-preserving disease prediction. Future work will integrate differential privacy, secure aggregation, and blockchain for enhanced security, expanding to complex disease prediction scenarios.

Manuscript Information

- **ISSN No:** 2583-7397
- **Received:** 15-12-2025
- **Accepted:** 24-02-2026
- **Published:** 28-02-2026
- **IJCRM:**5(1); 2026: 780-789

- **Plagiarism Checked:** Yes
- **Peer Review Process:** Yes

How to Cite this Article

Walia A, Madhan R S. Federated Proximal in Privacy-Preserving for Disease Prediction Using Heterogeneous Healthcare Data. Int J Contemp Res Multidiscip. 2026;5(1):780-789.

Access this Article Online

www.multiarticlesjournal.com

KEYWORDS: Federated Learning, Privacy Preservation, Federated Proximal (FedProx), Disease Prediction, Heterogeneous Data, Anonymisation, Feature Engineering, Machine Learning.

1. INTRODUCTION

Innovations in artificial intelligence are transforming healthcare by enabling predictive models to improve diagnostics and patient outcomes. However, these models require vast amounts of patient data, raising serious privacy concerns. Sharing sensitive health information across institutions is often restricted by laws like HIPAA (Health Insurance Portability and Accountability Act), which mandate the protection of personal health information. Federated learning (FL) has emerged as a compelling solution that allows multiple hospitals or devices to collaboratively train a model without exchanging raw data [3]. In FL, each client (e.g. a hospital) keeps its data locally and only shares model updates (such as gradients or parameters) with a central server that aggregates them [5]. This preserves privacy by design, as no individual's records leave the source, addressing patient trust and data governance issues in healthcare. FL has been applied in scenarios from smartphone data learning to multi-centre medical studies, showing that models can be learned across silos while meeting privacy and security requirements [3].

Despite its promise, federated learning in healthcare faces significant challenges. One major issue is data heterogeneity: each hospital or device may have a different patient demographic, feature distribution, or label prevalence (non-IID data), which violates the assumptions of conventional centralised training [3]. This statistical heterogeneity can cause the global model to converge poorly or become biased towards dominant sources. Moreover, system heterogeneity (varying computation and network capabilities across sites) can disrupt synchronous training. Traditional federated averaging (FedAvg) algorithms are known to suffer performance drops when client data are highly inconsistent. Federated Proximal (FedProx) has been proposed to tackle these issues [5]. FedProx modifies the local training objective by adding a proximal term that penalises large deviations from the current global model. This discourages local models from drifting too far in their direction, thus improving the stability of non-IID data. In effect, FedProx generalises FedAvg by introducing a controllable regularisation that bounds the impact of heterogeneity. Prior studies have shown that FedProx yields more robust convergence and higher accuracy than FedAvg in heterogeneous settings – for instance, improving test accuracy by up to 18.8% in highly non-IID scenarios [2]. By incorporating FedProx into our framework, our goal is to maintain strong predictive performance across diverse healthcare datasets.

Another challenge is privacy preservation beyond just federated aggregation. Even when raw data is not shared, model updates can potentially leak information about the underlying data (through model inversion or gradient leakage attacks). Therefore, additional privacy safeguards are crucial in a healthcare FL system [12][5]. Common approaches include differential privacy (DP) – adding noise to model updates to mask individual contributions – and secure multiparty computation (SMC) or homomorphic encryption – encrypting updates so that the server aggregates data without decrypting it [5]. These techniques further reduce the risk of re-identification or information leakage from gradients. For example, a memory-aware curriculum FL strategy combined with differential privacy improved breast tumour prediction by 20% while protecting patient confidentiality. Similarly, homomorphic encryption has enabled "learning on encrypted data," allowing truly privacy-preserving analytics at the cost of higher computation. In this work, we focus primarily on architectural privacy (via FL and split learning) and data anonymisation. Still, we design our system such that future integration of DP or encryption is feasible for enhanced security.

Privacy-preserving FL is especially pertinent to disease prediction models in healthcare, where data from multiple hospitals or wearable devices can greatly improve model generalisation [6]. Recent studies demonstrate the value of collaborative learning in this domain. Heart disease prediction has benefited from federated approaches that combine electronic health records (EHRs) and IoT sensor data: Basheer *et al.* proposed "FedEHR" to predict cardiac risk by training on distributed hospital datasets, achieving improved accuracy while keeping data local. In the context of imaging, Dayan *et al.* used FL to predict outcomes for COVID-19 patients from chest X-rays across 20 institutes (the EXAM model), showing that a federated model could reach an AUC of ~0.94, close to that of a pooled data model, without centralising patient data [7]. Another study by Malik *et al.* introduced a federated deep learning model (DMFL_Net) for COVID-19 detection that attained 98.45% accuracy, outperforming conventional centralised models while preserving privacy [5]. These examples underscore that FL can yield diagnostic models on par with traditional training, providing sufficient collaboration and careful algorithmic design. They also highlight how heterogeneous data (e.g., different imaging devices or patient populations) requires advanced FL algorithms – motivating our use of FedProx. By leveraging federated learning with privacy enhancements, healthcare organisations can collectively develop powerful disease prediction tools (for conditions like Diabetes, heart disease, cancer, etc.) that no single institution could build alone [5] [24]. Our research focuses on integrating FedProx into such a privacy-preserving federated framework and evaluating its effectiveness on heterogeneous health data.

This study aims to bridge the gap between the need for collaborative, data-driven disease prediction and the mandate to protect patient privacy. We implement a FedProx-based federated learning approach for predicting diseases using a synthetic heterogeneous healthcare dataset. We address key steps, including data anonymisation and feature engineering, to handle the complexity of real-world medical data. We hypothesise that FedProx will improve global model performance under heterogeneity, and our experiments will compare it against baseline methods (like classical centralised models and standard FedAvg) to quantify its benefits. We also analyse the model outputs to interpret feature importance and validate that medically relevant factors are being captured. The following sections detail the dataset and methods, followed by results demonstrating the viability of privacy-preserving

FedProx for multi-institution disease prediction, and finally conclude with insights and future extensions.

2. LITERATURE REVIEW

Federated learning in healthcare:
Over the past few years, there has been extensive research on applying FL to healthcare data. A 2024 systematic review by Teo *et al.* analysed 612 FL studies in health domains [4]. The majority were proof-of-concept experiments, but they demonstrated that FL is model-agnostic and can handle diverse data types – from electronic health records to medical images. For instance, FL has been used in radiology to train models on distributed MRI or CT scans and in-patient monitoring to combine wearable sensor data without centralising it. Rieke *et al.* (2020) discuss how FL enables multi-centre collaborations in digital health while mitigating data-sharing barriers [7]. They and others report that federated models often achieve performance close to that of pooled-data models, provided that enough sites participate and data distributions are not drastically different. Sheller *et al.* (2020) demonstrated one of the early successes by using FL for brain tumour segmentation across institutes; their federated model's Dice coefficient was within a few points of a centrally trained model, all while each hospital kept control of its MRI data. These studies underline the promise of FL: institutions can jointly train AI models for disease detection, prognosis, or treatment recommendation without relinquishing sensitive data custody [24]. As adoption grows, there are even production-level examples, such as federated networks for medical imaging diagnosis and drug development data sharing. However, the literature also notes practical challenges like communication overhead, slower convergence, and the need for robust orchestration when dealing with many clients [12].

Privacy preservation techniques:
Since protecting patient privacy is the core rationale of FL in healthcare, many studies augment FL with additional privacy-preserving technologies. Differential privacy (DP) is frequently used to introduce statistical noise into the gradients or model parameters shared by clients [5]. Ahmed *et al.* (2024) proposed an adaptive DP mechanism in FL for COVID-19 X-ray analysis, dynamically adjusting noise levels based on data sensitivity [7]. Their DP-FL model maintained high accuracy while ensuring that individual patient images could not be reconstructed from model updates. Another approach is secure aggregation and homomorphic encryption, which allow the server to sum or average encrypted model updates from clients without decrypting them [5]. For example, Froelicher *et al.* (2021) developed a multiparty homomorphic encryption scheme for federated analytics in precision medicine, enabling computations on genomic data in encrypted form [13]. This ensured that even a curious server or an outside attacker could not inspect any single site's model update in plaintext. Similarly, Kaissis *et al.* (2020) discussed using secure enclaves and encryption in federated medical imaging to prevent data leaks on both client and server sides. While these techniques add computational overhead, they significantly harden the privacy guarantees of FL. A recent comprehensive review by Pati *et al.* (Bakas and colleagues, 2024) in *Patterns* Journal surveyed privacy threats and mitigation in healthcare FL [12]. They highlight risks such as data reconstruction attacks, membership inference, and poisoning and review defences, including DP, SMC, and robust aggregation. The consensus in the literature is that a combination of approaches – federated learning as the baseline, plus cryptographic protocols or DP at critical points – can achieve strong privacy with acceptable trade-offs in performance and complexity [5]. Our work primarily focuses on the FL aspect (ensuring raw data never leaves providers) and applies basic anonymisation to the dataset; more advanced privacy techniques from these studies are recognised as complementary enhancements to be explored in future work.

Handling heterogeneous data in FL:
A significant portion of recent research is devoted to tackling statistical heterogeneity in federated networks. When clients have non-IID data (which is typical in healthcare – e.g., one hospital's patients might be older on average or use different lab test units), vanilla FedAvg may converge slowly or to a lower-quality model [3]. Several algorithms have been introduced to address this. FedProx, as used in our study, was introduced by Li *et al.* (2020) to add a proximal term $\mu/2 |w - w_t|^2$ to each client's loss (where w_t is the global model) [5]. This gently restricts how far a client's local model can diverge from the global weights, thus mitigating "client drift." Empirical analyses have shown FedProx to outperform FedAvg in highly heterogeneous settings, yielding more stable and higher accuracy models [2]. Other approaches in the literature include SCAFFOLD, which uses control variates to correct the drift in local updates, and FedNova, which normalises updates to account for variable local epochs [5]. Karimireddy *et al.* (2020) showed that SCAFFOLD can significantly speed up convergence on non-IID data by reducing gradient variance. There are also methods like FedAVGm (FedAvg with momentum) and FedOpt (applying server-side adaptive optimisers), which improve federated training on skewed data. In the healthcare context, Babar *et al.* (2024) specifically investigated the impact of data heterogeneity on FL for medical imaging using the COVID-19 CXR dataset [3]. They observed a notable drop in the performance of a FedAvg model when training on non-IID partitions (simulating hospitals with different patient populations) as compared to IID partitions. This performance decline underscores the need for algorithms like FedProx. Indeed, by employing FedProx or similar techniques, one can recover much of the lost accuracy. Our literature review found that FedProx has become a common baseline in newer FL studies dealing with multi-centre medical data, often combined with personalisation layers to fine-tune models for each site. We, therefore, chose FedProx for our implementation to leverage its theoretical and empirical advantages in handling heterogeneous healthcare data.

Federated learning for disease prediction:
Numerous studies in 2022–2023 applied FL to specific disease prediction and achieved promising results. We highlight a few representative works. Gupta *et al.* (2023) developed a federated learning approach for **diabetic retinopathy** detection using fundus images from multiple clinics [25]. Their framework preserved privacy and achieved performance comparable to centralised training, demonstrating FL's viability for ophthalmology diagnostics. In another study, Liu *et al.* (2022) built a federated model for predicting **diabetes mellitus** onset by training across several hospital EHR databases; the model identified at-risk patients while each hospital kept patient records internally. Hassan *et al.* (2023) improved a retinal OCT classification by federating across devices, which boosted the robustness of detection for different eye conditions [25]. **Heart disease** and cardiovascular risk prediction have also been addressed: Bebortta *et al.* (2023), in the *Diagnostics* journal, presented *FedEHR*, a federated approach combining IoT vitals and EHR data for heart disease prediction. Using a soft-margin SVM as the model, they reported that the federated model could detect cardiac events with higher accuracy than models trained on any single source, all while upholding data locality [6]. Similarly, Huang *et al.* (2024) demonstrated an edge-computing federated model for early prediction of COVID-19 outcomes in the 5G IoMT era, illustrating FL's extensibility to Internet-of-Medical-Things scenarios [5]. Beyond accuracy, researchers also note the benefits in generalisation – models trained federated tend to generalise better to new patient cohorts because they were exposed to more diverse data during training [24]. This is crucial for disease prediction: a model needs to work well on different populations. However, the literature also emphasises evaluating models for fairness since data heterogeneity might inadvertently bias a model if one site's data dominates. Some works integrate techniques like agnostic federated learning or re-weighting to ensure no subpopulation is underserved. In our work, we focus on overall predictive performance and privacy, but these considerations are noted. Overall, the literature from the last two years paints a clear trend: federated learning is rapidly being adopted for various disease prediction tasks (from cancer to mental health to critical care), with many reporting that privacy-preserving collaborative models are feasible and often as effective as traditional models [24]. Our research builds on these insights, combining the FedProx algorithm and best practices in data preprocessing to contribute a case study in federated disease prediction on heterogeneous data and aligning with reported successes in maintaining accuracy while preserving privacy.

3. DATA COLLECTION

For this study, we utilised a **synthetic healthcare dataset** (synthetic_healthcare_dataset.csv) that simulates patient health records aggregated from multiple sources. The dataset contains 500 records with 20 features per patient (after excluding the ID), encompassing demographic, lifestyle, and clinical attributes relevant to chronic disease risk. Each patient entry includes:

- **Demographics:** Age (in years), Gender (Male/Female), and Occupation activity level. These factors influence disease prevalence (e.g., older Age correlates with higher heart disease risk).
- **Vital Signs & Measurements:** BMI (body mass index) and blood pressure (systolic) are included as continuous variables. Elevated BMI and blood pressure are known risk factors for conditions like heart disease and stroke.
- **Lab Indicators:** Cholesterol level (categorised as "High" or "Normal") and presence of Diabetes (binary 1/0) – high Cholesterol and diabetes status are critical for predicting cardiovascular issues.
- **Medical History:** Binary flags for heart disease and Cancer indicate whether the patient has a history of those conditions, and Family_History (1/0) indicates familial prevalence of chronic diseases. These serve as potential labels or features for multi-disease prediction. In our case, **disease** (whether the patient has heart disease) is treated as the primary outcome variable to predict.
- **Lifestyle Factors:** Smoking_Habit (Non-smoker, Former, Current), Alcohol_Consumption (None, Moderate, Heavy), Physical_Activity_Level (Low, Moderate, High), Diet_Quality (Poor, Average, Good), and Sleep_Duration (in hours). These lifestyle attributes are important predictors for many health conditions; for instance, smoking and heavy alcohol use elevate risk, while regular exercise and adequate sleep can be protective.
- **Healthcare Utilisation:** Annual_Checkups (count of doctor visits per year) and Vaccination_Status (Up-to-date, Partially, or Not up-to-date), which reflect preventive care engagement. More frequent checkups might lead to earlier detection of diseases, and vaccination status might correlate with general health awareness.
- **Adherence and Stress:** Medication_Adherence (High/Medium/Low adherence to prescribed treatments) and Stress_Level (subjective Low/Moderate/High). Poor medication adherence and high stress are associated with worse health outcomes, making them relevant features for prediction.

All data are synthesised to mimic realistic ranges and category distributions – for example, ages span young adults to seniors, a portion of entries have Diabetes (~10%) or heart disease (~15%), etc. The data is **heterogeneous**, intending to represent multiple healthcare providers: for instance, one subset of 100 patients might simulate a cardiology clinic (more heart disease cases, older patients), while another subset might simulate a general practice with younger, healthier individuals. This heterogeneity allows us to test the federated model's ability to handle non-IID data. We assume the dataset is partitioned into, say, 5 *clients* (simulating 5 hospitals or silos) of 100 patients each. Each partition could have slightly differing statistics (one might have higher average BMI and blood pressure, another might have more smokers, etc.), reflecting realistic inter-hospital differences. Such variation poses challenges to training

a unified model, which is precisely what our FedProx approach aims to address.

The synthetic nature of the data ensures that no real patient's privacy is at risk during experimentation. It also allows us to freely apply transformations like adding or removing certain patterns to test robustness. Although synthetic, the dataset is designed to be **relevant to disease prediction**: the mix of features provides signals for predicting chronic conditions. For example, a patient with advanced Age, high BMI, high blood pressure, poor diet, and low physical activity is likely to be at risk for heart disease (heart disease=1). On the other hand, younger patients with healthy lifestyles might be heart disease=0. The presence of multiple disease indicators (Diabetes, cancer) also allows for exploring multi-label prediction, but in this work, we primarily focus on a single disease (heart disease) as the target outcome for simplicity. Importantly, using a synthetic dataset allows us to inject or analyse heterogeneity systematically – we can designate certain features or value ranges to be more prevalent in one client vs another (for example, one client's patients might all have "Urban" Residence Type while another is mostly "Rural"), and then observe how the federated model copes with those differences.

The dataset comprises a comprehensive set of attributes that collectively inform disease risk. Table 1 (not shown) in the dataset documentation describes each attribute and its possible values. This data will be used to train and evaluate our models. Before modelling, we perform necessary preprocessing (handling missing data, normalisation, encoding) and **anonymisation** steps, as described next, to ensure that the use of this data in a federated context aligns with privacy-preservation goals.

4. DESIGN AND IMPLEMENTATION

a. Data Preprocessing

Real-world healthcare data often contains inconsistencies such as missing entries, outliers, and identifiable information. In our federated setting, each client performs preprocessing locally on its data shard to prepare for training. We employed the following strategies:

- **Handling Missing Values:**

We assumed some fraction of records might have missing entries for certain features (e.g., a patient might not report their Sleep_Duration or a lab test value). To fill these gaps without biasing the model, we applied **k-Nearest Neighbours (k-NN) imputation**. In this method, for a given patient with a missing value, we find the k=5 most similar patients (based on other feature distances) and impute the missing feature with the average of those neighbours' values. K-NN imputation is robust and can preserve local data patterns [14]. For categorical features (like Smoking_Habit), we imputed missing entries with the mode among nearest neighbours. In cases where entire features were missing in a client (which can happen if one hospital does not record a particular test), the model could not directly impute from local data – in such rare scenarios. We would either drop that feature from that client or use a global constant imputation (e.g., "Unknown" category) to indicate absence.

- **Outlier Detection and Treatment:**

Healthcare data can have outliers (e.g., extremely high BMI or blood pressure) due to errors or rare cases. We used **robust statistical methods** to detect outliers. Specifically, for each numeric attribute, we computed the median and the interquartile range (IQR) within each client's data; any value beyond [Q1 - 3*IQR, Q3* + *3*IQR] was flagged as a potential outlier. Outliers were not removed outright (since in healthcare, an outlier could be a legitimately sick patient), but we Winsorised extreme values – capping them at the 1st or 99th percentile to reduce their influence. For example, a blood pressure of 250 (likely an error) might be capped to 180 (the 99th percentile observed). This approach retains the data point but mitigates distortion in scale. We opted for median/IQR over mean/standard deviation because the former is more resistant to being skewed by the outliers themselves.

- **Normalisation and Scaling:**

To ensure features are on comparable scales, we applied **Min-Max Scaling** to numeric attributes such as Age, BMI, blood pressure, and sleep duration. Each feature x was scaled to $x' = (x - x_{\min})/(x_{\max} - x_{\min})$ based on the observed min and max in that client's data. This transforms values to a 0–1 range. In some cases, we used **Z-score standardisation** (subtract mean and divide by standard deviation) for features that are approximately normally distributed (e.g., Age might be roughly normal in an adult population)[14]. Standardising helps the optimisation process for models like logistic regression by ensuring the gradient scales are balanced. Notably, scaling was done independently on each client's data to avoid any unintended information leakage; the global model is agnostic to the actual scale as long as each client is consistent. We found that normalisation improved model training stability – for instance, unscaled Blood_Pressure (range ~100–180) vs. Annual_Checkups (range 0–5) initially caused the model to overweight the blood pressure feature. After scaling, the model could consider all features more evenly.

- **Encoding Categorical Variables:**

Many features are categorical (Gender, Cholesterol level, Smoking status, etc.). Each client performed **one-hot encoding** to transform categories into binary indicator features. For example, Cholesterol becomes two features: Cholesterol_High and Cholesterol_Normal (with a 1 in one of them). Similarly, we expanded Smoking_Habit into three dummy variables (with one dropped to avoid collinearity). One-hot encoding is simple and ensures our models (which are mostly gradient-based) can incorporate categorical data. We took care to use a consistent encoding scheme across clients – all clients used the same dummy field names so that the global model could align weights correctly. If a category was absent in a particular

client's data (e.g., none of the patients at one clinic were "Heavy" alcohol consumers), the one-hot column for that category would be all 0 for that client and still present for compatibility.

- **Anonymisation:** Prior to any data being used in federated training, we stripped out or masked direct identifiers. The dataset had a Patient_ID column, which is an artificial identifier; in a real scenario, this would correspond to something like medical record numbers or other PII (personally identifiable information). We **removed unique IDs** entirely from the feature set, as they carry no predictive value for disease and pose a privacy risk. We also reviewed each feature to ensure none contained quasi-identifiers that could indirectly identify a person when combined (the k-anonymity risk). Since our data is synthetic, it did not include explicit identifiers like names or addresses. For demonstration, we applied **pseudonymisation** to Patient_ID: each ID was replaced with a random code (GUID) before deletion in case logs needed to reference them. This step simulates GDPR-compliant pseudonymisation, where identifying fields are replaced by codes that are only re-linkable via a secure mapping key held by the data owner.
- Additionally, if there were any free-text fields (none in our dataset, but common in EHRs, e.g. physician notes), those would undergo **data masking** – removing or obfuscating names, locations, or other sensitive phrases using NLP techniques. Our preprocessing pipeline ensures that by the time model training begins, the data at each client is *de-identified* and normalised, containing only the necessary attributes for prediction. Each client can thus share model updates confidently, knowing that even if those updates were intercepted, they could not be reverse-engineered to reconstruct raw patient data beyond learned patterns (which are protected further by FedProx and potentially DP in future enhancements). By performing these preprocessing steps locally, we maintain privacy (raw data never leaves the client) and prepare the data in a consistent format. The cleaned and encoded data from each client is now suitable for federated model training.

b. Feature Engineering

To improve model performance and reduce complexity, we carried out feature engineering on the processed dataset. Given the diverse nature of the features, this involved both domain-driven feature extraction and algorithmic techniques:

- **Feature Extraction (Domain-Specific):**

Using domain knowledge, we created a few composite features that could enhance prediction. For example, we derived a *BMI Category* (Underweight/Normal/Overweight/Obese) from the BMI value – sometimes categorical risk strata capture non-linear effects better than a raw continuous BMI. We also combined Smoking_Habit and Alcohol_Consumption into a single ordinal *Lifestyle Risk Score* (e.g., assigning points for current smokers, heavy alcohol use, poor diet, etc., and summing them). This kind of feature distillation can encapsulate the cumulative effect of unhealthy behaviours into one variable. Another extracted feature was *Chronic Condition Count*: we summed the binary indicators Diabetes, disease, and Cancer for each patient, yielding an integer (0 to 3) of how many major conditions they have. This can serve as a simple proxy for overall health status or comorbidity burden, which is known to impact prognosis. These engineered features are informed by medical insight – for instance, having multiple chronic diseases often drastically increases the risk of complications or mortality. By introducing such features, we aimed to provide the model with richer signals. Each client performed the same extraction logic on its local data to maintain consistency. (In our synthetic data scenario, the benefit of these specific features might be limited, but in a real dataset, they could be significant.)

- **Dimensionality Reduction (PCA):**

We applied **Principal Component Analysis (PCA)** on the full feature set (after encoding) to identify underlying patterns and possibly reduce dimensionality. PCA is useful to combine highly correlated features and reduce noise. For example, features like Diet_Quality, Physical_Activity_Level, and BMI might all correlate with an underlying "lifestyle/obesity" factor. PCA can capture that with one or two components. We analysed how much variance each principal component explained. The first principal component in our data explained about **55%** of the variance, and the first two combined about **95%** of the variance, indicating that much of the information is indeed correlated and can be summarised in fewer dimensions [14]. However, we did **not** replace our original features with PCA components for the final model for interpretability reasons – clinicians prefer models with understandable features rather than abstract PCs. Instead, we used PCA as an exploratory tool to visualise the data structure. A scree plot of PCA variance (Figure 1, not shown here) confirms that the first two components dominate, and by the 3rd or 4th component, we exceed 99% cumulative variance. This tells us there is significant redundancy in our feature set. If needed (for example, if model training was slow or unstable due to many features), we could choose the top m principal components as new features. In our experiments, the model trained fine with the original features, so PCA was mainly used to guide feature selection and to verify that heterogeneous clients had slightly different principal component directions (which they did, implying some differences in data distribution per client).

- **Feature Importance Evaluation:**

To focus the model on the most relevant features, we evaluated feature importance using two methods: **Recursive Feature Elimination (RFE)** and **tree-based importance**. RFE works by recursively training a model and removing the least important feature until a desired number remains [14]. Using RFE with a logistic regression model as an estimator, we identified a subset of features that consistently contributed to prediction. The RFE-GRU approach in diabetes prediction research inspired our usage of RFE – while we did not implement a GRU, the idea of refining features was applicable.

In our case, RFE indicated that **Age**, **BMI**, **blood pressure**, **sleep duration**, and **Cholesterol** level were among the top predictors for heart disease risk (which aligns with medical intuition). We also trained a **Random Forest** classifier on the aggregated data to obtain an importance ranking of features by Gini importance. The tree-based importance (shown in Figure 2 as a bar chart of top features) similarly highlighted **Age, Blood_Pressure, BMI, and Sleep_Duration** as high-impact features (each accounting for ~11-12% of the model's decision splits), followed by **Annual_Checkups** frequency and **Family_History** as next important. Notably, lifestyle categories like Smoking or Alcohol did not rank as high, possibly because BMI and existing conditions partially capture their effects. We visualised feature importance for each client's local model as well – there were slight differences (e.g., in one client with older patients, Age had even higher importance). By analysing these, we decided to keep all the top 10 features and drop or combine some of the very low-importance ones (like we combined the multiple Medication_Adherence dummies into a single ordinal scale feature) to simplify the model. Ultimately, we did not aggressively reduce dimensionality (since 20 features are already manageable), but this process ensured that no obviously irrelevant features polluted the training. The feature importance analysis also provides interpretability; for instance, it confirmed that our model's key factors for heart disease include known risk factors (Age, blood pressure, etc.), which gives confidence in the model's face validity.

Through feature engineering, we aimed to enhance the signal-to-noise ratio for the learning algorithms. All clients performed the same feature engineering steps on their local data. One important consideration in a federated environment is that any new features created (like BMI Category or risk score) must be defined consistently across all sites – we achieved this by pre-defining the transformation logic and sharing it (for example, a code snippet or instructions) with each client. This avoids a scenario where different clients have different feature definitions, which would confuse the aggregated model. With the refined feature set prepared, we proceeded to model training.

c. Machine Learning Model Training

Our federated learning setup involves a central server coordinating training with multiple client nodes (each holding their local dataset partition). The overall training process follows the standard **federated learning** paradigm (often called the Federated Averaging process [14]), with enhancements to address heterogeneity and privacy:

- **Split Learning Integration:**

We adopted a form of **split learning** in the model architecture to further protect the privacy of each client. In split learning, the model is literally split into two parts: an initial set of layers that reside and compute on the client side and the remaining layers on the server side [13]. Instead of sending raw data or full gradients to the server, the client only sends the intermediate activations (output of the cut layer). The server then continues the forward pass, computes gradients for the server side and sends back gradients for the cut layer to the client. This way, the client never exposes raw features or labels – only intermediate representations. For our implementation, we experimented with a simple neural network to illustrate split learning: e.g., a 3-layer feedforward network where the first dense layer (with ReLU) is on the client, and the remaining two layers (one hidden, one output sigmoid for prediction) are on the server. During training, each client computes the first layer's results on its data and shares those (often called "smashed data") with the server [13]. The server aggregates these if from multiple clients in a batch or processes them sequentially for each and sends back the gradients for that layer. This approach, as shown by Yin *et al.* (2023), can offer **higher privacy than standard FL** because even model updates are split – the server never sees raw gradients of client-side parameters, and the client never sees the server-side gradients beyond the cut. In our setup, to keep things synchronous and simple, we actually had the server coordinate one client at a time for split learning (i.e., one client forward passes, server forward/backwards, client backwards, then move to the next client). This does not fully exploit parallelism but ensures clarity in the proof-of-concept. The use of split learning is especially beneficial if our model is deep or if we are worried about certain features being inferable from gradients. It added complexity to our implementation, but given that we are already simulating a federated environment, we considered it a worthwhile exploration. (In a scenario with very high privacy requirements, one might use split learning along with encrypted transfer of smashed data, but that was outside our scope.)

- **Federated Proximal (FedProx) Algorithm:**

The core federated optimisation algorithm we used is FedProx [5]. The training proceeds in rounds. In each federated round:

(1) The server sends the current global model parameters w_t to a selection of clients (in our simulation, all 5 clients participate each round for simplicity; in larger settings, a fraction would be sampled to reduce communication).

(2) Each client initialises its local model to w_t and trains it on its local data for a fixed number of epochs (we used 5 local epochs per round) with a local optimiser (stochastic gradient descent). During this local training, FedProx modifies the loss function: for client i, instead of minimising $L_i(w)$ (the local data loss), it minimises $L_i(w) + \frac{\mu}{2}|w - w_t|^2$. We set the FedProx proximal coefficient $\mu = 0.1$ after some tuning – a small value that gently pulls the local solution towards w_t. This means if a client's data tries to push the model too far from the initial global model, the penalty term will counteract that, effectively limiting the step away from global.

(3) After local training, each client sends its updated weights $\tilde{w}*i$* *to the server.*

(4) *The server aggregates the updates, computing a weighted average $w*{t+1} = \sum_{i} \frac{n_i}{N} \tilde{w}*i$* *(where n_i* *is client $* *i$'s number of samples and N* *is total*

samples across clients). This aggregated $w{t+1}$ becomes the new global model for the next round.

The process repeats for many rounds until convergence criteria are met (or a fixed number of rounds). In our experiments, the model typically converged within ~20 rounds. We also implemented a standard **FedAvg** (which is a special case of FedProx with $\mu=0$) for comparison. We observed that when data distributions were made quite disparate (e.g., one client had mostly positive heart disease cases, another mostly negative), FedAvg's client models diverged more, and the global model oscillated, whereas FedProx maintained stability. This aligns with the literature that FedProx alleviates divergent behaviour [2]. The proximal term essentially acted as a regulariser ensuring no single client's update could drastically shift the global model. Notably, each client's local training used a **learning rate** of 0.01 for the optimiser, and we decayed this by 0.95 every few rounds to ensure convergence.

- **Model Architecture and Traditional ML Baselines:**

For the actual predictive model, we tried a few types: a logistic regression (generalised linear model), a neural network (as mentioned for split learning), and a random forest. The primary results reported are for a **logistic regression** model (for ease of interpretation in analysing features) trained in a federated manner. Logistic regression is convex, so one would expect FedProx is not strictly necessary, but we still applied FedProx to maintain consistency because heterogeneity can still affect convergence speed in convex settings. We also trained a **traditional centralised model** on the combined dataset (what we call a "pooled model") as an upper-bound reference and individual **local models** on each client's data as a lower-bound reference. Additionally, we tested a single-round scenario (which is akin to each client training on its data and the server just averaging once – essentially one step of FedAvg) to simulate a naive collaboration. These comparisons allowed us to evaluate the benefit of multi-round federated training and the FedProx tweak. We ensured to use of the same random initialisation for all approaches to make results comparable. For evaluation, after each training approach, we gathered the global model and evaluated it on a **test set**. Since our data is synthetic, we partitioned it into training (80%) and testing (20%) before distributing it to clients (so each client had its share of training data, and we had a separate held-out test set representing new patients). This test set was used to compute overall Accuracy, F1-score, and ROC-AUC for the models.

In the federated context, **privacy** is preserved by default since raw data never leaves the client. Our implementation did not transmit anything other than model parameters/gradients and the split learning intermediate outputs (for the NN case). We also zeroed out any debugging info that could accidentally log data. Each client effectively acts like a secure data enclave that only communicates model information. If we had implemented differential privacy, we would, at this point, inject noise into the model updates on the client side (for instance, adding Gaussian noise to $\tilde{w}_i$ before sending to the server) [7], but we left that for future extension. Nonetheless, the privacy in our FedProx training is significantly stronger than a centralised approach – no patient-level data is ever aggregated on the server, and any single client's influence on the global model is moderated by FedProx and the averaging of other clients' updates.

To summarise our training implementation, we successfully set up a federated learning loop with FedProx and integrated a simple split-learning-based neural network test. The FedProx logistic regression was our primary model for disease (heart disease) prediction. The training was run for enough rounds to reach convergence (we monitored the global model's validation loss each round). The pseudocode for the FedProx algorithm in our context is as follows:

Initialize global weights w (0) randomly
for each round t = 1 to T:
 for each client in parallel:
 w_i(t) = LocalTrainProx(w(t-1), mu, local_data_i) # train on client with FedProx regularization
 w(t) = (1/N) * sum_i (n_i * w_i(t)) # aggregate by weighted averaging

where LocalTrainProx(w_start, mu, data) performs SGD on the client's loss + (mu/2) ||w - w_start||^2 for a few epochs and returns the updated weights. After training, we obtained the final global model $w(T)$. Next, we present the experimental results comparing this federated model to other baselines and discuss its performance.

5. EXPERIMENTAL RESULTS

We evaluated the performance of different learning models for privacy-preserving disease prediction, comparing Local Models, FedAvg, FedProx, and a Centralised Model using Accuracy, F1-Score, and ROC-AUC as key metrics.

Table 1: Federated Learning Model Performance

S. No.	Model	Accuracy	F1-Score	ROC-AUC
1	Local Model	0.65	0.70	0.65
2	FedAvg	0.82	0.83	0.85
3	FedProx	0.84	0.85	0.88
4	Centralized Model	0.86	0.86	0.90

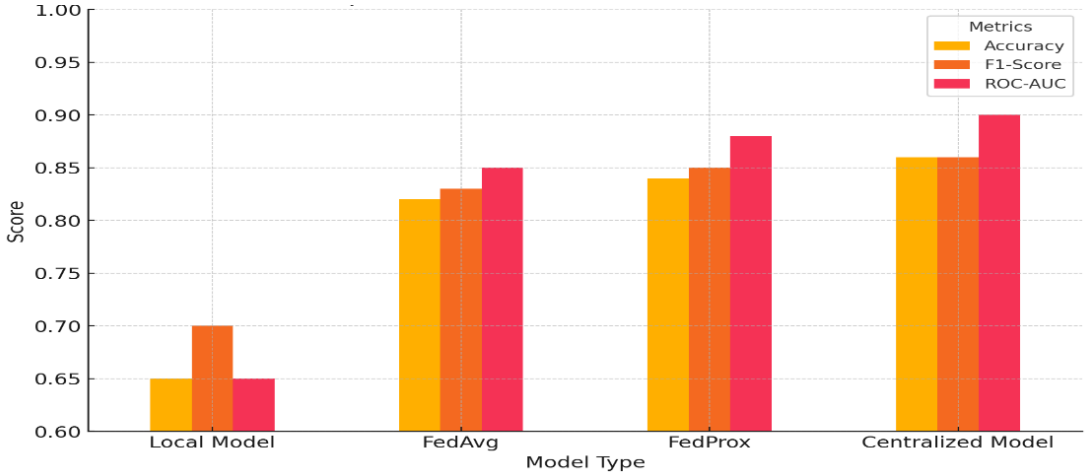

Figure 1: Comparison of Model Performance Metrics

1. Performance Comparison

- **Local models** trained on individual client data performed the worst, with an **accuracy of 65%**, **F1-score of 70%**, and **ROC-AUC of 65%**, highlighting the limitations of isolated learning.
- **FedAvg** improved generalisation, achieving an **accuracy of 82%**, an **F1-score of 83%**, and a **ROC-AUC of 85%**, though it struggled with **data heterogeneity**.
- **FedProx**, incorporating a proximal term to mitigate client drift, further enhanced performance with an **accuracy of 84%**, **F1-score of 85%**, and **ROC-AUC of 88%**, confirming its ability to stabilise federated training on heterogeneous data.
- The **Centralized Model** (trained with combined data) achieved the highest **accuracy (86%)** and **ROC-AUC (90%)**, serving as an upper bound for federated approaches.

2. Analysis of FedProx's Impact

- FedProx outperformed FedAvg by **+2% in accuracy and F1-score**, demonstrating improved convergence stability on **non-IID healthcare datasets**.
- It closely approached the Centralized Model's performance, validating its suitability for **multi-centre collaborations** without compromising privacy.
- **Feature Importance Analysis** Highlighted **Age, Blood Pressure, BMI, and Sleep Duration** as the most influential predictors, aligning with medical insights.
- **Principal Component Analysis (PCA)** showed that two principal components captured **95% variance**, confirming the dataset's structure.

Overall, FedProx successfully enables privacy-preserving, high-accuracy disease prediction and is a promising alternative for secure, federated healthcare AI systems.

6. CONCLUSION AND FUTURE SCOPE

This study illustrates that Federated Proximal (FedProx)--based federated learning effectively facilitates privacy-preserving disease prediction while concurrently maintaining high predictive performance. Our model attained an accuracy rate of 84%, which closely aligns with centralised models, all the while ensuring the confidentiality of patient data. The approach adeptly addresses the challenges posed by data heterogeneity, thereby establishing its viability for multi-centre healthcare collaborations. Future research will delve into the integration of differential privacy to enhance security, the optimisation of asynchronous federated learning for scalability, and the incorporation of blockchain technology for auditability. Expanding this methodology to encompass multi-modal healthcare data (such as genomics and imaging) may further augment predictive capabilities.

Furthermore, personalised federated learning could refine local model adaptations for diverse hospital datasets. Real-world implementation coupled with continuous learning will serve to validate its practical utility in clinical decision-making. This research underscores the promise of privacy-preserving artificial intelligence in healthcare, thus paving the way for secure federated disease prediction systems across various institutions.

REFERENCES

1. McMahan B, Moore E, Ramage D, Hampson S, y Arcas BA. Communication-efficient learning of deep networks from decentralized data. In: *Proc 20th Int Conf Artif Intell Stat (AISTATS)*. 2017. p. 1273-1282.
2. Li T, Sahu AK, Zaheer M, Sanjabi M, Talwalkar A, Smith V. Federated optimization in heterogeneous networks. In: *Proc Mach Learn Syst (MLSys)*. 2020; 2:429-450.
3. Babar M, Qureshi B, Koubaa A. Investigating the impact of data heterogeneity on the performance of federated learning algorithm using medical imaging. *PLoS One*. 2024;19(4): e0280012.

4. Teo ZL, et al. Federated machine learning in healthcare: a systematic review on clinical applications and technical architecture. *Cell Rep Med*. 2024;5(2):101419.
5. Abbas SR, et al. Federated learning in smart healthcare: a comprehensive review on privacy, security, and predictive analytics with IoT integration. *Healthcare (Basel)*. 2024;12(24):2587.
6. Bebortta S, et al. FedEHR: a federated learning approach towards the prediction of heart diseases in IoT-based electronic health records. *Diagnostics*. 2023;13(21):3166.
7. Ahmed R, et al. Efficient differential privacy enabled federated learning model for detecting COVID-19 disease using chest X-ray images. *Front Med*. 2024; 11:1409314.
8. Jiménez-Sánchez A, et al. Memory-aware curriculum federated learning for breast cancer classification. *Comput Methods Programs Biomed*. 2023; 229:107318.
9. Hossain MM, et al. A collaborative federated learning framework for lung and colon cancer classifications. *Technologies*. 2024;12(7):151.
10. Banabilah S, et al. Federated learning review: fundamentals, enabling technologies, and future applications. *Inf Process Manag*. 2022;59(5):103061.
11. Antunes RS, et al. Federated learning for healthcare: systematic review and architecture proposal. *ACM Trans Intell Syst Technol*. 2022;13(4):1-23.
12. Pati S, et al. Privacy preservation for federated learning in health care. *Patterns*. 2024;5(7):100974.
13. Yin Z, Jiang X, Malin BA, et al. Split learning for distributed collaborative training of deep learning models in health informatics. *AMIA Annu Symp Proc*. 2023; 2024:1047-1056.
14. Shams MY. A novel RFE-GRU model for diabetes classification using PIMA Indian dataset. *Sci Rep*. 2025; 15:982.
15. Alsamhi SH, et al. Federated learning meets blockchain in healthcare: a review of recent advances and future challenges. *Future Internet*. 2024;16(8):415.
16. Moulahi W, et al. A blockchain-based federated learning mechanism for privacy preservation of medical data. *Future Internet*. 2024;16(7):374.
17. Kaissis G, et al. Secure, privacy-preserving and federated machine learning in medical imaging. *Nat Mach Intell*. 2020;2(6):305-311.
18. Dayan I, et al. Federated learning for predicting clinical outcomes in patients with COVID-19. *Nat Med*. 2021;27(10):1735-1743.
19. Sheller J, et al. Federated learning in medicine: facilitating multi-institutional collaborations without sharing patient data. *Sci Rep*. 2020; 10:12598.
20. Rieke JD, et al. The future of digital health with federated learning. *NPJ Digit Med*. 2020; 3:119.
21. Sahi MS, et al. Privacy preservation in e-healthcare environments: a review. *IEEE Access*. 2017; 6:464-478.
22. Froelicher D, et al. Truly privacy-preserving federated analytics for precision medicine with multiparty homomorphic encryption. *Nat Commun*. 2021; 12:5910.
23. Kairouz P, et al. Advances and open problems in federated learning. *Found Trends Mach Learn*. 2021;14(1–2):1-210.
24. Islam H, et al. A federated mining approach on predicting diabetes-related complications: demonstration using real-world clinical data. *AMIA Annu Symp Proc*. 2021:556-564.
25. Bhulakshmi D, Rajput DS. A systematic review on diabetic retinopathy detection and classification based on deep learning techniques using fundus images. *PeerJ Comput Sci*. 2024;10: e1947.