AN EXTENDED SCAFFOLD TECHNIQUE BASED ON COSINE SIMILARITY FOR CLIENT DRIFT MITIGATION IN CROSS-SILO FEDERATED LEARNING

James Mburu Muthii

A Thesis Submitted in Partial Fulfilment of the Requirements for the Degree of Master of Science in Computer Science of Murang'a University of Technology

August, 2026

DECLARATION

Student Declaration

I hereby declare that this thesis is my own original work and has not been submitted, in whole or in part, for any degree or examination at any other institution. All sources of information and literature used in this work have been duly acknowledged and referenced. Any contributions made by others have been explicitly stated. I confirm that the research presented herein was conducted in accordance with the ethical guidelines and academic integrity standards of my institution.

_____ _____

Name: James Mburu Muthii Date

Reg No: SC400/5143/2023

Supervisor's Approval

The undersigned have read and recommend for acceptance in Murang'a University of Technology a thesis entitled: **"An Extended Scaffold Technique Based on Cosine Similarity for Client Drift Mitigation in Cross-Silo Federated Learning".**

_____ _____

Dr. Stephen Kahara Wanjau Date

Department of Computer Science

Murang'a University of Technology

_____ _____

Dr. Stephen Njenga Date

Department of Computer Science

Murang'a University of Technology

DEDICATION

This work is affectionately dedicated to my beloved parents, Mr. Peter Mburu and Mrs. Lucy Ngure, and to my sibling, Esther Wangu. Thank you for your unwavering mental, physical, and financial support throughout this journey. Your constant belief in my potential and your endless sacrifices provided me with everything I needed to reach this milestone. I am forever grateful.

ACKNOWLEDGMENT

I would like to express my deepest gratitude to everyone who made the completion of this thesis possible through their academic, logistical, and personal support. First and foremost, I am profoundly grateful to my primary supervisor, Dr. Stephen Kahara, for his invaluable guidance, immense patience, and insightful critiques throughout the course of this research. I also extend my sincere thanks to my secondary supervisor, Dr. Stephen Njenga, whose constructive feedback and academic expertise greatly helped shape this work

I am highly indebted to the leadership of the School of Computing and Information Technology (SCIT). I express my gratitude to the Chairman of the Department (COD) of Computer Science, Dr. Aaron Oirere, and the Dean of SCIT, Dr. John Ndia, for providing a conducive environment and the necessary administrative support required for my studies. Special thanks go to Prof. Geoffrey Muchiri, Director of the Board of Postgraduate Studies (BPS), and Dr. Gabriel Kamau, Director of the Open, Distance and e-Learning (ODel) directorate, for their institutional support, oversight, and efficiency in facilitating my postgraduate journey.

I would also like to thank my peer and labmate, Shadrack Sendeu, for generously providing me with dedicated space to work and essential IT support, which kept my research moving forward smoothly during critical phases.

ABSTRACT

Federated learning enables multiple clients to collaboratively train a shared model without exchanging raw data, yet its effectiveness under non-independent and non-identically distributed (non-IID) data is undermined by client drift, where repeated local updates pull each client away from the global objective. Existing mitigations, notably Scaffold, correct for drift using control variates but apply a fixed, uniform local learning rate to every client regardless of whether its update direction agrees with or opposes the prevailing global trajectory -a directional signal that a PRISMA 2020-compliant review of 33 studies (screened from 1,847 records) confirmed remains markedly underexplored in the literature. This gap motivated the development of CosScaffold (Cosine Similarity-extended Scaffold), a novel algorithm that, unlike prior approaches, modulates each client's effective learning rate by the cosine similarity between its gradient-direction probe and the previous round's aggregated server update, amplifying well-aligned updates and attenuating misaligned ones through a clipped affine modulator that gracefully recovers vanilla Scaffold when directional information is absent. Implemented in PyTorch and benchmarked against six baselines on EMNIST Balanced and CIFAR-10 under a pre-registered protocol, CosScaffold reached target accuracy in a median of 16.0 communication rounds at the most heterogeneous setting -17.9% fewer rounds than Scaffold (19.5) and 5.9% fewer than FedDyn (17.0) -while achieving the highest final test accuracy of 0.498 against Scaffold's 0.480, a 3.75% relative improvement. A construct-validity analysis of 1,960 paired observations further confirmed cosine and Euclidean alignment to be statistically distinct ($R^2 < 0.01$), substantiating the novelty of the directional signal exploited. CosScaffold thus contributes a lightweight, communication-free, theoretically grounded mechanism for exploiting gradient alignment in federated optimisation, offering both a new algorithmic paradigm for non-IID learning and a practical, drop-in enhancement that accelerates convergence precisely where data heterogeneity is most severe.

TABLE OF CONTENTS

DECLARATION ..... **ii**
DEDICATION ..... **iii**
ACKNOWLEDGMENT ..... **iv**
ABSTRACT ..... **v**
LIST OF TABLES ..... **x**
LIST OF FIGURES ..... **xi**
ACRONYMS AND ABBREVIATIONS ..... **xii**
DEFINITION OF TERMS ..... **xiv**
CHAPTER ONE: INTRODUCTION **1**
1.1 Background of the Study 1
1.2 Statement of the Problem 3
1.3 Research Objectives 4
1.3.1 General Objective 4
1.3.2 Specific Objectives 5
1.4 Research Questions 5
1.5 Significance of the Study 5
1.6 Scope 7
1.7 Limitations 9
1.8 Contributions of the Thesis 13
1.8.1 Contributions to Algorithmic Design and Empirical Understanding 13
1.8.2 Contribution to Practice 15
1.9 Organisation of the Thesis 16
CHAPTER TWO: LITERATURE REVIEW **18**
2.1 Introduction 18
2.2 Overview of Federated Learning 18
2.2.1 The Federated Learning Process 18
2.2.2 Key Characteristics of Federated Learning 19
2.2.3 Categories of Federated Learning by Data Partitioning 20
2.2.4 Categories of Federated Learning by Scale Setting 23
2.2.5 Key Challenges in Federated Learning 23
2.2.6 Applications of Federated Learning 24
2.3 Client Drift in Federated Learning 25
2.3.1 The Non-IID Data Problem 25
2.3.2 Understanding Client Drift 27
2.4 Theoretical Review 29
2.4.1 Optimization Theory in Distributed Systems 29
2.4.2 Federated Learning (FL) and FedAvg 31

2.4.3 Stochastic Controlled Averaging (Scaffold) 32
2.4.4 Cosine Similarity Theory 34
2.5 Existing Techniques for Client Drift Mitigation 36
2.5.1 Aggregation-Based Methods 36
2.5.2 Regularization-Based Methods 36
2.5.3 Variance Reduction via Control Variates 37
2.5.4 Dynamic Regularization 38
2.5.5 Parameter-Level Drift Correction 38
2.5.6 Personalization-Based Methods 39
2.5.7 Knowledge Distillation Approaches 39
2.5.8 Comparative Summary 40
2.6 Design Paradigms and Theoretical Foundations for Client Drift Mitigation 44
2.6.1 Variance Reduction and Control Variates 44
2.6.2 Proximal Operator Theory 45
2.6.3 ADMM and Primal-Dual Optimisation 45
2.6.4 Meta-Learning 46
2.6.5 Contrastive Learning 46
2.6.6 Knowledge Distillation 46
2.6.7 Feedback Control Theory 47
2.6.8 Gradient Normalisation and Momentum 47
2.7 Results of the Systematic Literature Review (Objective 1) 48
2.7.1 SLR Search Strategy and Protocol 48
2.7.2 Approaches Identified for Extending Scaffold (RQ1) 51
2.7.3 Limitations of Existing Approaches (RQ2) 54
2.7.4 Scaffold Characteristics Extended (RQ3) 56
2.7.5 Key Findings and Research Gap 56
2.8 Existing Evaluation Techniques and Benchmarks 58
2.8.1 Evaluation Metrics 58
2.8.2 Benchmarks 60
2.8.3 Validation Techniques in Federated Learning 62
2.9 Chapter Summary 64
CHAPTER THREE: METHODOLOGY 68
3.1 Introduction 68
3.2 Research Design 69
3.3 Research Process and Phases 70
3.4 Design of the Enhanced Scaffold Technique (Objective 2) 74
3.4.1 Cosine Similarity-based Control Mechanism 74
3.4.2 Formulation of Adaptive Learning Rates and Variates 75
3.5 Evaluation Framework 80

3.5.1 Experimental Setup and Simulation Environment....81
3.5.2 Selection of Benchmark Datasets....84
3.5.3 Simulating Data Heterogeneity....87
3.6 Implementation Details....88
3.6.1 Execution environment and Pipeline....88
3.6.2 Data Preparation....89
3.6.3 Heterogenity Application....90
3.6.4 Hyperparameter Tuning....90
3.7 Data Analysis....90
3.7.1 Performance Metrics....91
3.7.2 Descriptive Analysis....94
3.7.3 Inferential Analysis....94
3.7.4 Stable-Window Classifier....96
3.8 Reliability and Validity....97
3.9 Ethical Considerations....100
3.10 Chapter Summary....102
CHAPTER FOUR: RESULTS AND DISCUSSION....105
4.1 Introduction....105
4.2 Design of the Enhanced Scaffold Technique....106
4.2.1 Conceptual Design....106
4.2.2 Mathematical Formulation....107
4.2.3 Algorithmic Framework....109
4.3 Performance Validation and Comparison....111
4.3.1 Convergence Rate Analysis....111
4.3.2 Final Model Accuracy....113
4.3.3 Impact of Statistical Heterogeneity....114
4.3.4 Computational Overhead....117
4.3.5 Trajectory Analysis....118
4.4 Construct Validity: Cosine versus Euclidean Alignment....122
4.5 Discussion of Findings....124
4.5.1 Directional Alignment versus Magnitude Correction....124
4.5.2 Theoretical Alignment....125
4.5.3 Magnitude and Practical Significance of the Observed Gains....126
4.5.4 Practical Implications....127
4.5.5 Limitations of the Technique and Conditions....129
4.6 Chapter Summary....131
CHAPTER FIVE: SUMMARY, CONCLUSION AND RECOMMENDATIONS....133
5.1 Summary of Research Findings....133

5.2 Conclusion 136
5.3 Recommendations for Practice and Policy 137
5.4 Suggestions for Further Research 140
5.4.1 Full-Scale Replication and Statistical Confirmation 140
5.4.2 The Euclidean Variant 141
5.4.3 Extension to Larger and More Diverse Testbeds 142
5.4.4 Cross-Device Federated Settings and Real-World Deployment 142
5.4.5 Formal Convergence Analysis 143
REFERENCES 144
APPENDICES 154

LIST OF TABLES

Table 2.1: Comparative Summary of Client Drift Techniques. 43
Table 2.2: Distribution of Included Studies across the Seven Digital Libraries. 51
Table 2.3: Scaffold Extension Approaches for Client Drift Mitigation. 53
Table 2.4: Scaffold Characteristics Extended for Client Drift Mitigation. 58
Table 2.5: Chapter Summary 67
Table 3.1: Phases of the Research Process and their Mapping to Research Objectives. 74
Table 3.2: Five-Arm Ablation Framework for Isolating the Cosine Modulator's Contribution. 81
Table 3.3: The Seven Algorithms Compared in this Study 85
Table 3.4: Experimental Configurations Spanning Convex and Non-Convex Regimes 88
Table 3.5: Similarity Levels Used to Simulate Data Heterogeneity 90
Table 3.6: Tuned Hyperparameters for Configuration A (EMNIST and Logistic Regression). 93
Table 3.7: Performance Metric Families Tracked Across Each Experimental Run. 95
Table 3.8: Pre-registered Hypothesis Families and Statistical Tests. 98
Table 4.1: Mean Final Test Accuracy after 50 Communication Rounds on Configuration A, by Algorithm and Similarity Level. 116
Table 4.2: F4 Construct-Validity Analysis: Correlation between Cosine Similarity and Euclidean Distance across 1,960 (Round, Client) Paired Observations from the Controlled-Scale Evaluation. 125

LIST OF FIGURES

Figure 2.1: Federated Learning Process. 19
Figure 2.2: HFL Category of Federated Learning by Data Partitioning. 21
Figure 2.3: VFL Category of Federated Learning by Data Partitioning. 22
Figure 2.4: FTL Category of Federated Learning by Data Partitioning. 23
Figure 2.5: IID vs Non-IID Data Distribution Across Clients in Federated Learning. 27
Figure 2.6: IID vs Non-IID Data Distribution Across Clients in Federated Learning. 28
Figure 2.7: Client Drift Under IID (a) and Non-IID (b) Data Conditions 30
Figure 2.8: Client Drift Under IID (a) and Non-IID (b) Data Conditions 31
Figure 2.9: PRISMA 2020 Flow Diagram for Selection and Evaluation of Studies. 52
Figure 4.1: Communication Rounds Required to Reach the Pre-registered Target Accuracy on Configuration A (EMNIST and Logistic Regression), across the Three Similarity Levels. 114
Figure 4.2: Convergence Stability of the Three Top-Performing Algorithms across the Three Similarity Levels on Configuration A. 118
Figure 4.3: Stable-Window Classification across all Seven Algorithms on Configuration A, at each of the Three Similarity Levels. 119
Figure 4.4: Test-Loss Trajectories on Log-Log Axes for all Seven Algorithms 122

ACRONYMS AND ABBREVIATIONS

FL	Federated Learning
FedAvg	Federated Averaging
IID	Independent and Identically Distributed
Non-IID	Non-independent and Identically distributed
Scaffold	Stochastic Controlled Averaging for Federated Learning
CosScaffold	Cosine Similarity extended Scaffold technique
CNN	Convolutional Neural Networks
ResNet	Residual Network
NeurIPS	Conference on Neural Information Processing Systems
ICML	International Conference on Machine Learning
SGD	Stochastic Gradient Descent
ADMM	Alternating Direction Method of Multipliers
API	Application Programming Interface
BGD	Bounded Gradient Dissimilarity
BHD	Bounded Hessian Dissimilarity
CCPA	California Consumer Privacy Act
CIFAR	Canadian Institute For Advanced Research
CUDA	Compute Unified Device Architecture
DoI	Digital Object Identifier
EMD	Earth Mover's Distance
ERM	Empirical Risk Minimization
FedDC	Federated Learning with Local Drift Correction
FedDF	Federated Ensemble Distillation Framework
FedDyn	Federated Learning Based on Dynamic Regularization
FedMA	Federated Learning with Matched Averaging
FedNova	Federated Normalized Averaging
FedNTD	Federated Not-True Distillation
FedProx	Federated Proximal
FTL	Federated Transfer Learning
GAM	Gradient Norm-Aware Minimization
GDPR	General Data Protection Regulation
GPU	Graphics Processing Unit

HFL	Horizontal Federated Learning
ICLR	International Conference on Learning Representations
KD	Knowledge Distillation
KL	Kullback-Leibler
MAML	Model-Agnostic Meta-Learning
MLP	Multi-Layer Perceptron
MMSE	Minimum Mean Square Error
MOON	Model-Contrastive Federated Learning
PID	Proportional-Integral-Derivative
SAM	Sharpness-Aware Minimization
SAGA	Stochastic Average Gradient Algorithm
SARAH	Stochastic Recursive gradient algoritHm
ScaffoldComp	Scaffold with Gradient Compression
SHA	Secure Hash Algorithm
SLR	Systematic Literature Review
SVRG	Stochastic Variance Reduced Gradient
VFL	Vertical Federated Learning
VRAM	Video Random Access Memory

DEFINITION OF TERMS

Ablation Study -An experimental procedure that isolates the contribution of a specific algorithmic component by systematically removing or replacing it and comparing performance against the full system. Used in this thesis via a five-arm framework to isolate the effect of the cosine modulator.

Bounded Gradient Dissimilarity (G, B) -A theoretical assumption used in the analysis of federated optimisation algorithms, positing constants G and B that bound the average squared local gradient norm relative to the global gradient norm, thereby quantifying the degree of statistical heterogeneity across clients.

Client Drift -The phenomenon in which a client's local model parameters progressively diverge toward a local optimum during multiple local training steps, moving away from the global objective under non-IID data conditions.

Communication Round -A single cycle in federated learning in which the server broadcasts the global model to sampled clients, clients perform local training, and the server aggregates the resulting updates

Construct Validity -The extent to which a measurement instrument (e.g., cosine similarity) captures the theoretical concept it is intended to represent, rather than an unrelated or overlapping property.

Control Variate -A variance-reduction technique, borrowed from Monte Carlo estimation, that corrects a biased or noisy estimator using a correlated quantity of known mean; in Scaffold and CosScaffold, control variates correct the discrepancy between local and global gradient directions.

Cosine Similarity -A measure of directional alignment between two non-zero vectors, computed as the normalised inner product; bounded in [-1, +1] and invariant to vector magnitude. Used in this thesis to gauge the alignment between a client's gradient probe and the previous round's aggregated server update.

CosScaffold (Cosine Similarity extended Scaffold) -The federated learning algorithm proposed in this thesis, which extends Scaffold by introducing a per-client, cosine-similarity-based multiplicative modulator that adaptively scales the local learning rate.

Cross-Silo Federated Learning -A federated learning setting involving a relatively small number (typically 2–100) of reliable, well-resourced, and stateful organisational clients, such as hospitals or banks; the setting adopted in this study.

Data Heterogeneity (Non-IID Data) -The condition in which client datasets are not independently and identically distributed, differing in label distribution, feature characteristics, or sample size, a central challenge in federated learning.

Dirichlet Partitioning -A method of simulating non-IID data by drawing each client's label proportions from a Dirichlet distribution parameterised by a concentration value α; identified in this thesis as an alternative to the Hsu-Qi-Brown scheme actually used, and proposed as future work.

Effective Local Step Size (η_eff) -In CosScaffold, the local learning rate actually applied by a client after multiplication by its cosine-derived modulator, as opposed to the fixed nominal step size used in standard Scaffold.

FedAvg (Federated Averaging) -The foundational federated learning algorithm in which clients perform local stochastic gradient descent and the server aggregates updates via a weighted average.

FedDyn -A federated learning algorithm that mitigates client drift through dynamic regularisation of the local objective using a per-client dual variable, used as a comparator baseline in this study.

FedProx -A federated learning algorithm that constrains client drift by adding a proximal regularisation term penalising deviation from the global model; used as a comparator baseline in this study.

Federated Learning (FL) -A distributed machine learning paradigm in which multiple clients collaboratively train a shared model under server coordination without exchanging raw data.

Gradient Probe -In CosScaffold, a single stochastic gradient evaluated at the current server model by a client at the start of a round, used solely to estimate the client's local update direction for the cosine similarity computation.

Hsu-Qi-Brown Partitioning Scheme -The non-IID data partitioning method (also used in the original Scaffold study) in which a percentage of each client's data is sampled IID and the remainder allocated by label-sorted slicing, controlled by a single similarity parameter.

Identity-Clamping -The design property whereby the CosScaffold modulator is fixed to a value of 1 whenever directional information is unavailable (e.g., the first round or near-zero vector norms), ensuring the algorithm reduces exactly to standard Scaffold under these conditions.

Modulator (m_i) -The per-client multiplicative scaling factor, derived from cosine similarity, that adjusts the effective local learning rate in CosScaffold.

Non-IID Data -See Data Heterogeneity.

Pre-registration -The practice of publicly or cryptographically committing to a study's hypotheses, design, and analysis plan before data collection, used in this thesis to prevent post-hoc selection of favourable results.

Scaffold (Stochastic Controlled Averaging for Federated Learning) -A federated learning algorithm that mitigates client drift using server- and client-level control variates to correct gradient-level bias; the direct predecessor extended by this thesis

ScaffoldComp -A compressed variant of Scaffold combining control variates with top-k sparsification and error feedback to reduce upload communication cost; used as a comparator baseline in this study.

Stable-Window Classifier -A pre-registered diagnostic tool developed in this thesis that classifies a training run's terminal behaviour (converged, still learning, or non-converged failure) based on the standard deviation and slope of test accuracy over the final rounds

Statistical Heterogeneity -See Data Heterogeneity.

Variance Reduction -A family of optimisation techniques that lower the variance of a gradient estimator without introducing bias, typically using a correlated reference quantity; the theoretical foundation underlying Scaffold and CosScaffold.

Weight Divergence -The $\ell 2$-norm distance between a client's locally trained model parameters and the global model, used as an empirical indicator of client drift severity.

CHAPTER ONE
INTRODUCTION

1.1 Background of the Study

Federated Learning (FL) is a distributed machine learning paradigm in which multiple clients, such as mobile devices or institutional servers, collaboratively train a shared global model under the coordination of a central server, without any client sharing its raw data [1]. The dominant technique, Federated Averaging (FedAvg), broadcasts a global model to a subset of clients each round, lets them perform several steps of local stochastic gradient descent on their private datasets, and then aggregates the resulting model updates on the server through a weighted average [1]. This design makes FL particularly attractive in domains where data privacy is paramount, including healthcare, finance, and mobile computing. However, FedAvg depends heavily on the assumption that client data is approximately independent and identically distributed (IID) , an assumption that rarely holds in practice. Real-world client datasets differ substantially in their label distributions, feature characteristics, and sample sizes, giving rise to what is commonly termed the non-IID setting.

Under non-IID conditions, each client's local objective function can differ markedly from the global objective. When a client performs multiple local gradient steps between communication rounds, its model parameters are progressively pulled towards a local optimum that may lie far from the optimum of the aggregate loss. This accumulation of directional error across local steps is known as client drift [2]. Zhao,et al. [3] provided early empirical evidence of the problem, showing accuracy losses of up to 55% on highly non-IID partitions relative to IID baselines. Karimireddy,et al. [2] later formalised these observations, proving tight convergence bounds for FedAvg that identified client drift as the principal bottleneck in heterogeneous federated settings. Its severity scales with both the number of local update steps and the degree of statistical heterogeneity across clients, making it a persistent and practically significant challenge.

Several methods have been proposed to address client drift. FedProx [4] adds a proximal regularisation term that penalises large deviations from the global model during local training, while FedDyn [5] and FedDC [6] use dynamic regularisation and auxiliary drift variables, respectively, to narrow the gap between local and global

model trajectories. Although these approaches offer measurable improvements, they act indirectly, constraining or regularising local updates rather than correcting the gradient-level drift itself. Scaffold (Stochastic Controlled Averaging for Federated Learning), proposed by [2], takes a more direct approach by introducing control variates borrowed from the variance reduction literature. Each client maintains a local control variate that estimates its own update direction, while the server maintains a global one. The difference between the two estimates is applied as a correction term during local training, steering each client's gradient steps back towards the global objective. Scaffold converges at a rate independent of the degree of data heterogeneity and, unlike FedAvg, benefits from additional local computation steps [2].

Despite these strengths, standard Scaffold treats all clients uniformly during local training: every client applies the corrected gradient using the same local learning rate, regardless of whether its update direction is already well aligned with the global one or has drifted sharply away from it. This uniform update strategy can be unnecessarily aggressive when alignment is strong, allowing well-aligned clients to overshoot, or insufficiently restrained when alignment is weak, allowing misaligned clients to contribute updates that aggregation will largely cancel. Cosine similarity, a metric that quantifies the directional agreement between two vectors on a scale from -1 to $+1$, has recently gained traction in the federated learning community for tasks such as client selection [7], Byzantine-robust aggregation [8], and gradient-based privacy defence [9]. Its sensitivity to directional alignment rather than magnitude makes it a natural candidate for gauging the severity of client drift in real time. As FL is increasingly deployed in large-scale, heterogeneous environments where the nature of drift varies across clients and training stages, the need for adaptive drift correction has become both timely and practically important.

To address this gap, the present study developed an extended Scaffold technique that uses cosine similarity as an adaptive control mechanism for the local update. Rather than altering Scaffold's control-variate correction itself, the proposed method modulates the local learning rate at which each client applies its corrected gradient, scaling it according to how well the client's local update direction agrees with the previous round's aggregated server update. Clients whose directions agree with the most recent global trajectory have their local step amplified, since they contribute

information broadly consistent with the global objective. Clients whose directions oppose it have their local step attenuated, since aggregation would otherwise partially cancel their contributions. A key design property is that the mechanism reduces exactly to standard Scaffold whenever no directional information is available, such as in the first communication round or when a client's gradient probe carries no usable signal. The approach was therefore designed to preserve Scaffold's convergence guarantees while improving practical performance under severe, heterogeneous data distributions.

1.2 Statement of the Problem

Client drift remains a central obstacle to efficient and stable convergence in federated learning under non-IID data conditions [2], [3]. Several techniques have been developed to mitigate it, yet each carries notable limitations that recent comparative and empirical studies have made explicit, in addition to the theoretical concerns already documented. FedProx [4] constrains local updates through a proximal term that penalises deviation from the global model, but this penalty is isotropic: it treats every direction of deviation alike and does not distinguish drift that harms the global objective from drift that reflects genuinely useful local information. This limitation is not merely theoretical: in a large-scale comparative benchmark spanning multiple non-IID partitioning strategies and datasets,[32] found that non-IID data brings significant challenges to the learning accuracy of federated learning algorithms and that none of the state-of-the-art methods evaluated outperforms the others across all settings. More specifically, the same study reports that FedProx tracks FedAvg closely throughout training in many cases, since the regularisation strength that performs best is typically small enough that the proximal term has little practical influence on the outcome [32] - direct experimental confirmation that FedProx's indirect, isotropic constraint frequently fails to translate into a measurable drift-correction benefit over the uncorrected baseline it was designed to improve upon. FedDyn [5] attempts a more targeted correction through dynamic regularisation, yet it relies on assumptions that limit its applicability at scale, including a high rate of client resampling and prior knowledge of the total number of clients [10]. In cross-device settings where thousands of devices participate intermittently, these assumptions are difficult to satisfy, and the unbounded growth of FedDyn's internal state variables has been shown to cause asymptotic instability [10], with empirical trajectories showing FedDyn's

auxiliary state norm and test loss diverging together over communication rounds precisely in the large-scale regime where it is meant to be most useful. FedDC [6] introduces auxiliary variables to track local–global parameter discrepancies, but its correction operates at the parameter level rather than the gradient level, so it addresses a symptom of drift rather than its root cause: the directional divergence of local gradients from the global update trajectory.

Scaffold [2] represents the most principled response to client drift to date, directly estimating and correcting the gradient-level discrepancy through control variates. However, the local learning rate at which each client applies the corrected gradient is fixed and uniform across all clients and all local steps within a communication round, so the local step cannot respond to the changing geometry of the loss landscape as the model traverses parameter space during training. The same step size is applied whether the client's current update direction is nearly aligned with the global direction, where a smaller step would prevent overshoot, or sharply divergent from it, where a smaller step would limit contributions that aggregation will largely cancel. This static behaviour becomes increasingly problematic as the number of local steps grows, precisely the regime where Scaffold is otherwise expected to excel and where objective inconsistency between clients performing unequal amounts of local work has been shown to bias the aggregated update away from the true global optimum [23]. Its practical fragility under realistic deployment conditions has also been observed directly: under partial client participation, Scaffold has been shown to fail to work effectively at all [32], a regime that is the norm rather than the exception in the large-scale, cross-device deployments this study targets. Taken together, these comparative findings indicate that the shortfall of existing mitigation techniques is not confined to theoretical critique but is reproducible across independent experimental settings, reinforcing the need for a mechanism that adapts its correction to each client's directional alignment rather than applying a fixed rule uniformly.

1.3 Research Objectives

1.3.1 General Objective

To develop an extended Scaffold technique based on Cosine Similarity for Client Drift mitigation in Cross-Silo Federated Learning.

1.3.2 Specific Objectives

i. To investigate factors of Scaffold for client drift mitigation in Federated Learning.
ii. To develop an extended Scaffold technique based on the identified factor(s) for client drift mitigation in Federated Learning.
iii. To evaluate the performance of the extended Scaffold technique against baseline techniques using simulated image-classification benchmarks under varying data heterogeneity.

1.4 Research Questions

i. What are the extendable factor(s) in the existing approaches, characteristics and their limitations used in extending Scaffold for client drift mitigation in Federated Learning?
ii. How to develop an extended Scaffold technique based on the identified factor(s) for client drift mitigation in Federated Learning?
iii. How to evaluate the performance of the extended Scaffold technique against baseline techniques using simulated image-classification benchmarks under varying data heterogeneity?

1.5 Significance of the Study

This study holds significance for both the advancement of theoretical knowledge in federated learning and the practical improvement of distributed machine learning systems deployed in privacy-sensitive environments. Its relevance is set out below in terms of its theoretical contributions, practical applications, and the specific groups that stand to benefit from its findings:

From a theoretical standpoint, this study addresses a gap in the federated learning literature concerning the adaptivity of drift correction mechanisms within the Scaffold framework. While Scaffold [2] introduced control variates as a principled approach to client drift, its uniform application of the local learning rate across all clients and all local steps does not account for the varying severity of drift at different points during training. Through a PRISMA-compliant [12], [13] systematic literature review of 33 studies across seven databases, this research established that similarity-based approaches for modulating drift correction remain significantly underexplored, with

only one prior study employing any similarity measure in this context. The study is therefore significant in introducing a novel theoretical mechanism, cosine similarity between local and global update directions, as an adaptive scaling factor for the local learning rate, extending understanding of how directional alignment information can be used to improve variance reduction in federated optimisation. The empirical findings further contribute new evidence that client drift is not a static property of the data partition but a dynamic characteristic that evolves over the course of training, an insight that enriches the theoretical understanding of drift beyond the static bounds established in prior analytical work.

From a practical standpoint, the significance of this study lies in the potential of the CosScaffold (Cosine Similarity extended Scaffold) technique to reduce the number of communication rounds required to train accurate global models in federated settings with heterogeneous data. In real-world deployments, each communication round between the server and participating clients carries tangible costs in bandwidth, time, and energy consumption. Fewer rounds to reach a target accuracy level translate directly into faster model deployment, lower operational expenditure, and reduced demand on network infrastructure. This is particularly relevant for federated learning systems operating in bandwidth-constrained environments, such as mobile networks in developing regions, or in latency-sensitive contexts such as intercontinental hospital consortia training diagnostic imaging models. The technique's design as a drop-in - meaning that it can replace the existing method without requiring substantial modifications to the surrounding system while extending its functionality - enhancement to existing Scaffold implementations, requiring only three additional hyperparameters with intuitive defaults and no changes to the communication protocol, makes it readily adoptable by practitioners without significant re-engineering effort.

The findings are expected to benefit several groups. Federated learning researchers working on optimisation techniques for heterogeneous settings can draw on the systematic review's catalogue of nine extension approaches, twelve extendable Scaffold characteristics, and their documented limitations as a foundation for future investigations. Machine learning engineers and system architects deploying federated learning pipelines in production can use the CosScaffold (Cosine Similarity extended

Scaffold) technique and its evaluation results to make informed decisions about drift correction strategies suited to their particular data landscape and infrastructure constraints. Organisations in privacy-sensitive sectors, including healthcare institutions collaborating on diagnostic models without sharing patient records, financial institutions jointly developing fraud-detection classifiers, and mobile computing platforms refining on-device models through federated training, stand to benefit from faster and more stable convergence of their collaborative models. Ultimately, the end users of these systems, patients whose diagnoses are informed by collaboratively trained medical models, consumers protected by jointly developed security systems, and mobile device users served by improved on-device applications, are the downstream beneficiaries of the improved efficiency and reliability that this research seeks to deliver.

1.6 Scope

This study was situated within the domain of federated learning optimisation, specifically targeting the problem of client drift that arises when training is conducted over statistically heterogeneous (non-IID) client data distributions. The scope encompassed the design, implementation, and empirical evaluation of CosScaffold (Cosine Similarity extended Scaffold), an extended Scaffold technique that uses cosine similarity as an adaptive control mechanism, modulating each client's local learning rate according to how well its local update direction agrees with the previous round's aggregated server update.

In terms of algorithmic focus, the study was concerned exclusively with gradient-direction-aware modulation of the local step within the Scaffold framework. The extension introduced a per-client multiplicative modulator derived from the cosine similarity between a probe of the client's local gradient direction and the previous round's aggregated server update, clipped to a bounded interval and clamped to the identity when no directional information was available. By design, this construction reduces exactly to vanilla Scaffold in the no-information regime, preserving the convergence properties of the original technique. The study compared CosScaffold against six baselines, FedAvg, FedProx, Scaffold, FedDyn, and ScaffoldComp, all implemented from scratch within a single shared training harness to remove cross-algorithm implementation differences as a confound. Other drift mitigation techniques

such as FedDC, MOON, and personalisation-based methods were reviewed in the literature to contextualise the problem but were not included as experimental baselines, since the primary aim was to evaluate the specific contribution of cosine-similarity-based modulation of the local step relative to the uniform step size of standard Scaffold.

With respect to datasets and tasks, the study was scoped to three experimental configurations spanning convex and non-convex regimes. Configuration A used logistic regression on the EMNIST Balanced dataset as a convex testbed comprising 47 classes of handwritten digits and letters. Configuration C used a two-layer multilayer perceptron on the same dataset as a small non-convex testbed. Configuration D used a small convolutional neural network on CIFAR-10 as a non-convex image-classification testbed. EMNIST Balanced was selected because it is the dataset used by [2] in their primary empirical evaluation of Scaffold, enabling direct comparison with the published baseline; CIFAR-10 was retained as a more challenging non-convex extension that the original Scaffold paper did not address. The study did not extend to tasks beyond image classification, such as natural language processing, speech recognition, or recommendation systems, nor to larger architectures such as ResNets or transformers.

Non-IID data heterogeneity was simulated using the Hsu-Qi-Brown partitioning scheme, which is also the scheme employed by [2] in their non-IID experiments. Under this scheme, a single similarity parameter controls the degree of heterogeneity: a percentage of each client's data is sampled IID from the full training set, and the remainder is allocated by label-sorted slicing across clients. Three similarity levels were used: 0% (fully heterogeneous, worst-case non-IID), 10% (mostly heterogeneous with mild IID overlap), and 100% (fully IID, used as an upper bound). The study did not extend to other forms of heterogeneity such as feature distribution skew, quantity skew, or temporal distribution shift, focusing instead on label distribution skew, the most commonly studied and well-understood source of client drift, and the one that most directly aligns with the heterogeneity regime in the published Scaffold baseline.

The federated learning setting was a simulated cross-silo environment with 100 clients and a sampling fraction of 20% per round, each sampled client performing five local

epochs of stochastic gradient descent using a batch fraction of 20% of its local data. The experiments assumed a synchronous communication protocol with a single central server and did not consider asynchronous federated settings, decentralised (peer-to-peer) topologies, or cross-device scenarios involving thousands or millions of clients. All experiments were run in a single-process PyTorch simulation, with the 100 clients simulated sequentially rather than distributed across machines; the study therefore did not address practical deployment concerns such as communication latency, bandwidth constraints, client availability, or device heterogeneity in computational capacity.

The evaluation was scoped to four metric families that collectively assess convergence (rounds to reach target accuracy, final test accuracy, test loss trajectory), communication cost (cumulative bytes uploaded to reach target accuracy), drift diagnostics (per-round Euclidean distance between client updates and the aggregated server update, and cosine similarity between the client probe direction and the previous aggregated update), and convergence stability (standard deviation and slope of the test-accuracy trajectory over the final twenty rounds, used to classify each run as converged, still learning, or non-converged failure, with a failure-mode sub-classification). To isolate the contribution of the cosine modulator and rule out alternative explanations, the study also specified a five-arm ablation framework comparing the full proposed algorithm against an identity arm (recovering vanilla Scaffold), a uniformly random modulator, a sign-only modulator, and an anti-modulator with opposite sign. The study did not include formal convergence proofs or theoretical analysis of CosScaffold; it was primarily an empirical investigation under controlled conditions, with all hypotheses, statistical tests, and analytical thresholds pre-registered before data collection. It also did not address privacy-preserving mechanisms such as differential privacy or secure aggregation, since these are orthogonal concerns that operate independently of the drift correction mechanism under investigation.

1.7 Limitations

While this study was designed to provide a rigorous empirical evaluation of CosScaffold, several limitations arising from its methodology and design choices constrain the generalisability and interpretability of its findings. These limitations are

acknowledged below, together with the specific measures taken to address each so that their impact on the reliability of the results was minimised:

First, the reliance on simulation rather than real-world federated deployment represents a fundamental methodological constraint. All experiments were run within a single-process simulation in which the server-client communication pattern was modelled programmatically rather than occurring over actual network infrastructure. Consequently, the study could not capture the effects of communication latency, packet loss, variable bandwidth, or the stochastic availability of client devices that characterise production federated learning systems. This limitation was overcome by reporting communication cost analytically in cumulative bytes uploaded per algorithm rather than in wall-clock time, so that the comparative findings remain interpretable independent of any particular network environment.

Second, the study was limited to two real-world benchmark datasets, EMNIST Balanced and CIFAR-10, supplemented by a synthetic quadratic testbed. The original methodology had also specified Fashion-MNIST under Dirichlet partitioning as a third real-world testbed; this was substituted at implementation time to keep the controlled-scale evaluation within the available hardware budget, with the substitution and its rationale recorded in Chapter Four. The datasets retained are standard in the federated learning literature and enable direct comparison with the published Scaffold baseline, but they do not fully represent the complexity, scale, or diversity of data encountered in real-world federated applications such as medical imaging or financial fraud detection. A related limitation attaches to the partition scheme used to induce non-IID heterogeneity. The label-sorted-with-IID-mixing scheme of [84], adopted here because it is the scheme used by the original Scaffold paper, captures label-distribution skew but not feature-distribution skew, quantity skew, or temporal distribution shift, which can co-occur in production federated deployments. The behaviour of the cosine modulator under these alternative heterogeneity regimes is therefore not characterised by the present study. This limitation was overcome by spanning three distinct experimental configurations covering convex (logistic regression) and non-convex (MLP and CNN) regimes across two model scales, providing breadth across function classes even within a constrained dataset selection, and by identifying the extension

to Fashion-MNIST under Dirichlet partitioning, and to non-image domains more generally, as a priority future-work item in Subsection 5.4.3.

Third, the convolutional and fully-connected architectures employed were modest in depth and parameter count, and it remains an open question whether the benefits of cosine-similarity-based modulation would persist for substantially larger models with millions or billions of parameters, where the geometry of the loss landscape and the behaviour of gradient directions may differ qualitatively. This limitation was overcome by selecting architectures consistent with the model-size constraints discussed by [2], ensuring the findings are at minimum directly comparable to the established Scaffold baseline at the same scale.

Fourth, the non-IID data heterogeneity was induced using a single partitioning strategy, the Hsu-Qi-Brown scheme of [84], controlled by a similarity parameter applied to label distributions, the same scheme used by the original Scaffold paper and therefore the most directly comparable choice. This limitation was overcome by evaluating across three similarity levels spanning the full label-skew spectrum, from fully non-IID through mildly mixed to fully IID, thereby characterising the algorithm's behaviour across the full range of label-skew severity; the orthogonal heterogeneity dimensions not captured by this scheme are discussed under the dataset limitation above.

Fifth, the number of participating clients was fixed at 100 with a sampling fraction of 20% per round, reflecting a cross-silo federated environment. Cross-device scenarios, which may involve thousands or millions of intermittently available clients with highly variable data quality and computational capacity, present a substantially different optimisation landscape. The behaviour of cosine similarity as a directional indicator may change in such settings, where the server's aggregated direction is updated from a very small fraction of the total client population each round. This limitation was overcome by fixing the client population and sampling fraction to values that match the established Scaffold evaluation protocol, ensuring that the comparative findings against Scaffold are not confounded by differences in federation scale.

Sixth, the study did not provide formal theoretical convergence guarantees for CosScaffold. Standard Scaffold benefits from rigorous convergence proofs that establish its independence from the gradient dissimilarity parameter, and introducing a data-dependent modulator into the update rule alters the theoretical properties of the technique in ways that were not formally analysed. Two complementary design choices were made to bound the consequences of this gap. First, the modulator is defined with identity-clamping, so that CosScaffold reduces exactly to vanilla Scaffold whenever no directional information is available, ensuring that the convergence guarantees of the original technique are preserved at the boundary conditions and that the proposed algorithm is a strict generalisation rather than a replacement. Second, in the active state of the modulator (that is, when the cosine signal is informative and the modulator departs from the identity), the per-client effective step size is bounded above and below by the fixed clip values, $\eta_{min} = 0.5$ and $\eta_{max} = 1.5$, multiplied by the underlying Scaffold local learning rate η_l. The effective step-size envelope under CosScaffold is therefore contained within a fixed multiplicative neighbourhood of the Scaffold step-size envelope; any choice of η_l that lies inside the convergence regime of standard Scaffold also induces a CosScaffold step-size that lies inside the same regime, scaled by a factor in $[\eta_{min}, \eta_{max}]$. This is an informal argument rather than a formal proof, since the convergence rate of Scaffold (Theorem III of [2]) was derived for a uniform step size, and substituting a per-client, per-round-bounded step size is not in general a benign modification. Deriving a formal convergence rate for CosScaffold in its active state, ideally one that exposes the dependence on the modulator hyperparameters, is identified as a priority future-work item in Subsection 5.4.5.

Seventh, the modulator was defined as a clipped affine function with fixed coefficients for the slope and the lower and upper clip bounds. While these values were chosen to enforce a conservative modulation envelope, they were not exhaustively optimised through systematic hyperparameter search, and alternative formulations could yield stronger performance gains. This limitation was overcome by pre-registering the modulator coefficients before any experimental data was generated and by complementing the main analysis with a five-arm ablation framework that isolates the contribution of the modulator's direction and magnitude, ensuring that any observed

benefit is attributable to the specific design choice rather than to incidental hyperparameter tuning.

Eighth, the computational overhead introduced by CosScaffold, while modest, was not negligible. The technique requires computing a gradient probe and a cosine similarity at the start of each client's participation in a round. This limitation was overcome by computing the probe as a single stochastic gradient at the server model (a one-pass operation) rather than a full-batch quantity, and by clamping the modulator to the identity when norms fall below a numerical floor, ensuring that the additional overhead remained a small constant fraction of the standard Scaffold computation.
Finally, simulation-based experimentation inherently raises concerns about reproducibility and the temptation to adjust analytical decisions post hoc in light of observed results. This limitation was overcome through a cryptographically anchored pre-registration in which the comparator set, modulator hyperparameters, ablation arms, hypothesis families, statistical procedures, stable-window classifier thresholds, and convergence criteria were frozen in a single JSON document before any data was collected, with its SHA-256 digest embedded in every per-run summary file. Any examiner or independent researcher can recompute this digest to verify that the analysis reported in this thesis was specified in advance of data collection, providing a stronger reproducibility guarantee than conventional pre-registration platforms.

1.8 Contributions of the Thesis

This study makes several contributions to both the theoretical knowledge base of federated learning and the practical advancement of distributed machine learning systems. The contributions are organised below in line with the three specific objectives of the research.

1.8.1 Contributions to Algorithmic Design and Empirical Understanding

This study conducted a PRISMA [12] 2020-compliant systematic literature review (SLR) across seven electronic databases, from which 33 of 1,847 identified records were included, reinforced by expert opinion and analysis of the SLR's recommendations. The review produced the first dedicated systematic catalogue of Scaffold extensions for client drift mitigation, identifying nine distinct extension approaches and twelve Scaffold characteristics targeted for modification. It revealed

that variance reduction dominates the field (eleven of thirty-two empirical studies, 34%; Table 2.2), while similarity-based approaches, communication mechanisms, and privacy budget allocation remain significantly underexplored. Most critically, only one of thirty-two studies employed a similarity measure (Pearson correlation) for drift mitigation, and that study itself acknowledged that it could not distinguish healthy from malicious similarity. Expert analysis of this gap led to the evidence-based recommendation of cosine similarity as an adaptive control variate, providing the theoretical foundation for the CosScaffold (Cosine Similarity extended Scaffold) technique developed in Objective Two.

The principal algorithmic contribution is the CosScaffold (Cosine Similarity extended Scaffold) technique, which extends Scaffold's update rule by adding a per-client multiplicative modulator that scales the effective local learning rate according to the cosine similarity between the client's gradient direction probe and the previous round's aggregated server update. The design advances the study of variance reduction in federated optimisation in three ways. First, it uses cosine similarity as a computationally inexpensive proxy for instantaneous drift severity that acts on the local step size rather than on the control variate correction, a mechanism not previously realised within the Scaffold framework. Second, the modulator is bounded by a clipping interval and clamped to the identity in the no-information regime, embodying a transferable design principle: directional information should refine the standard update where it is reliably available, and the technique should fall back to the established baseline where it is not. Third, CosScaffold is constructed as a strict generalisation of standard Scaffold, recovering the original technique whenever the directional signal is unavailable or the modulator is set to the identity, so that Scaffold's convergence guarantees are retained at these boundary conditions; a formal analysis of the regime in which the modulator is non-trivially active is identified as future work (Section 5.5.5).

The evaluation employed four metric families: convergence (rounds to reach target accuracy, final test accuracy, test loss trajectory), communication cost (cumulative bytes uploaded per algorithm), drift diagnostics (per-round Euclidean distance between client updates and the aggregated server update, alongside cosine similarity between client gradient probes and the previous aggregated update), and convergence

stability (standard deviation and slope of the test-accuracy trajectory over the final twenty rounds, used to classify each run as converged, still learning, or non-converged failure, with a failure-mode sub-classification), one of the more multifaceted assessments of a Scaffold-based extension in the current literature. This breadth substantiates the claim that adaptive modulation of the local step offers measurable benefits over uniform stepping under non-IID conditions. Notably, tracking the modulator's evolution across rounds adds a novel interpretability dimension: by following how the cosine similarity signal and the resulting per-client modulator values respond from round to round, the study provides direct empirical evidence that drift severity is not a static property of the data partition but an evolving characteristic of the training process. This finding extends the empirical understanding of client drift beyond the static bounds established in prior work and suggests that future drift correction methods should account for this temporal variability.

1.8.2 Contribution to Practice

The SLR and its expert-reinforced analysis serve as a practitioner-oriented reference guide for selecting drift mitigation strategies matched to specific deployment constraints. Engineers in privacy-sensitive domains such as healthcare, finance, and mobile computing can consult the catalogue of approaches and their limitations to make informed algorithmic choices based on their communication budget, client participation patterns, and data heterogeneity profile. The review's identification of underexplored characteristics offers a research roadmap, directing future effort towards areas with the greatest potential for real-world impact. The PRISMA-compliant protocol and quality assessment framework also constitute a reusable methodological template for systematic reviews in adjacent areas of federated learning.

CosScaffold provides a drop-in enhancement to existing Scaffold implementations, introducing only three additional hyperparameters: the modulator slope α, which controls how strongly the cosine signal influences the effective local step, and the lower and upper clipping bounds η_{min} and η_{max}, which constrain the modulator to a safe envelope around the identity. All three have intuitive defaults under which the modulator behaves as a conservative multiplicative refinement of the standard local step, and the technique requires no additional communication bandwidth and no

changes to the server-client protocol beyond what standard Scaffold already requires. For organisations training models across distributed data silos, such as hospital consortia developing diagnostic models or financial institutions building fraud-detection classifiers, CosScaffold enables faster and more stable convergence, so that deployment-ready accuracy is reached in fewer communication rounds. This translates directly into reduced time-to-deployment, lower cumulative communication cost, and faster delivery of improved AI services to end users, all without compromising the data privacy guarantees that motivate federated learning.

The evaluation results provide directional empirical evidence, a controlled-scale proof-of-concept that practitioners can draw on when considering adoption of CosScaffold. The communication-efficiency findings are especially significant: fewer rounds to reach target accuracy translate directly into savings in bandwidth, time, and energy in deployed federated systems, whether across mobile networks in developing regions or intercontinental hospital networks. The convergence-stability results reduce the risk of deploying a model checkpoint that coincides with a performance dip, improving the reliability of federated systems in production. For end users, patients relying on collaboratively trained diagnostic models, or consumers protected by jointly developed fraud-detection systems, the impact is a faster path to more accurate AI services without compromising data privacy. The evaluation framework itself, comprising four metric families plus a stable-window classifier and a controlled experimental protocol under non-IID conditions, is a reusable contribution for benchmarking future federated optimisation techniques.

1.9 Organisation of the Thesis

This thesis is organised into five chapters. Chapter One has introduced the study, situating client drift within the broader landscape of federated learning, stating the problem, articulating the research objectives and questions that follow from it, and delimiting the scope, limitations, and contributions of the work. Chapter Two reviews the underpinning literature. It surveys the federated learning paradigm and its challenges, develops the theoretical apparatus through which client drift is formalised, examines the existing techniques and design paradigms applied to the problem, and presents a PRISMA 2020-compliant systematic literature review of 33 studies that identifies the specific gap in the use of similarity-based mechanisms within the

Scaffold framework, thereby satisfying Objective One. Chapter Three sets out the research methodology. It states the research design, walks through the six phases by which the study was conducted, presents the CosScaffold (Cosine Similarity extended Scaffold) technique in full mathematical detail, and specifies the evaluation framework, data-analysis plan, reliability and validity safeguards, and ethical considerations governing the empirical work, thereby satisfying Objective Two and laying the groundwork for Objective Three. Chapter Four reports the results of the controlled-scale evaluation and discusses their implications. It presents the experimental setup, the headline rounds-to-target and final-accuracy comparisons against six baseline algorithms across three similarity levels, the convergence-stability analysis, and the construct-validity analysis that distinguishes cosine similarity from Euclidean distance as alignment measures, thereby satisfying Objective Three. Chapter Five concludes the thesis. It summarises the principal findings, states the study's overall conclusion, sets out recommendations for practice and policy, and identifies the most informative directions for further research, including the full-scale replication that the cryptographically pre-registered protocol is designed to support. The thesis closes with the list of references cited throughout and the appendices documenting the pre-registration JSON, the manifest of experimental artefacts, and the supporting computational protocols.

CHAPTER TWO
LITERATURE REVIEW

2.1 Introduction

This chapter sets out the theoretical, methodological, and empirical context for the CosScaffold (Cosine Similarity extended Scaffold) technique developed in this thesis. It addresses Objective One: to investigate the approaches, characteristics, and limitations of techniques that extend Scaffold for client-drift mitigation in federated learning, and so identify the factor on which a principled enhancement can rest. Where the objective requires it, the review adopts PRISMA 2020-compliant systematic-review rigour [12], [13], providing the basis for the design choices in Chapter Three.

The chapter moves from a general overview to systematic analysis. Section 2.2 introduces federated learning and its principal challenges [1], [14], [15]; Section 2.3 examines client drift, the central optimisation pathology arising under heterogeneous data [1], [2], [3]. Sections 2.4 to 2.6 develop the theoretical apparatus, survey existing drift-mitigation techniques, and review their design paradigms. Section 2.7 reports the systematic literature review for Objective One, Section 2.8 reviews the evaluation metrics and benchmarks, and Section 2.9 sets out the research gap that motivates CosScaffold.

2.2 Overview of Federated Learning

2.2.1 The Federated Learning Process

The canonical FL process, instantiated by FedAvg [1], runs over communication rounds. Each round, the server selects a subset of available clients and broadcasts the current global model. Each client initialises its local model, runs one or more epochs of stochastic gradient descent (SGD) on its local data, and returns the difference between its updated and received models, the model update or pseudo-gradient. The server aggregates these updates, typically as a weighted average proportional to each client's dataset size, forming a new global model for the next round, repeating until convergence or a stopping criterion is met [1], [15].

This differs from conventional data-centre training in defining ways. There, data is shuffled and partitioned uniformly across workers over a fast, reliable network, with every worker participating each step [15]. In FL, the data partition follows the natural

distribution across clients and cannot be controlled; the network may be slow, costly, or unreliable; only some clients are available per round; and clients differ widely in compute and dataset size [1], [16]. These are not incidental but defining characteristics shaping algorithmic design throughout FL. Figure 2.1 illustrates the canonical process.

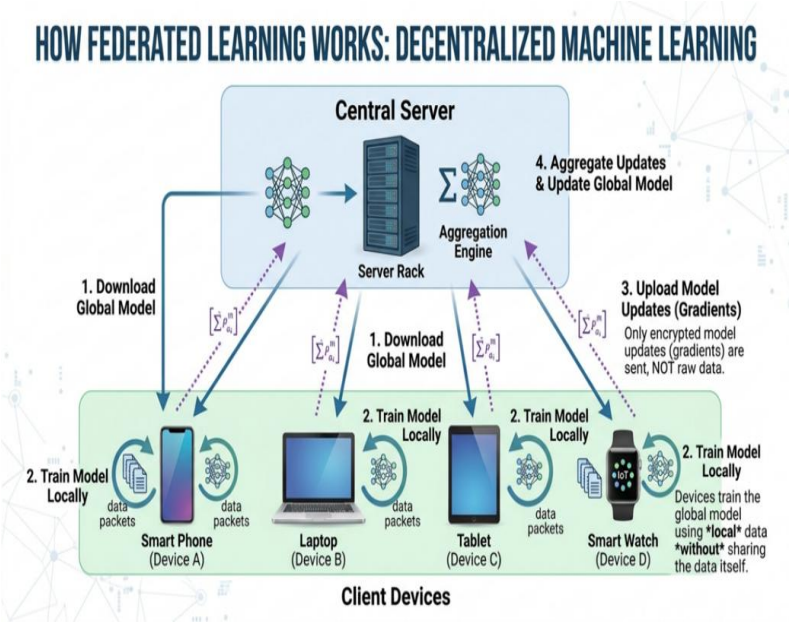


Figure 2.1: Federated Learning Process.

2.2.2 Key Characteristics of Federated Learning

Several characteristics distinguish FL from conventional distributed learning and drive its central challenges:

i. Data locality and privacy. Raw data never leaves the client; only model updates such as gradients or parameter deltas are shared [1], [14]. This makes FL attractive for privacy-sensitive applications, though it does not guarantee formal privacy, since updates can still leak information about the data [17].

ii. Statistical heterogeneity (non-IID data). Unlike centralised training, where data can be shuffled and sampled uniformly, each client's data reflects its own usage, location, protocols, or demographics, so distributions are typically non-independent and non-identically distributed (non-IID) [3], [15]. Zhao et al. [3] showed this can cut accuracy by up to 55% relative to IID baselines, making it one of FL's most consequential challenges.

iii. Systems heterogeneity. Clients differ greatly in compute, memory, bandwidth, and battery life [4], [16]; devices in the same federation can differ by orders of

magnitude in their capacity to train and transmit, affecting the speed and reliability of participation.

iv. Communication constraints. Communication, not computation, is often the main bottleneck [1], [15]. Cross-device clients use limited, high-latency mobile networks, and even cross-silo participants transmit large parameter sets each round. Reducing communication rounds, or compressing per-round information, is therefore central to FL design [16].

v. Partial and unreliable participation. Not all clients are always available: cross-device clients may join only when charging, on Wi-Fi, and idle [15], while cross-silo partners may drop out through failures or policy. FL methods must therefore tolerate partial participation and stragglers, clients that take disproportionately long to compute [1], [4].

2.2.3 Categories of Federated Learning by Data Partitioning

Yang et al. [14] classify FL by how data is distributed across the feature and sample spaces, yielding three categories: horizontal, vertical, and federated transfer learning.

i. Horizontal Federated Learning (HFL), or sample-based FL, applies when clients share the same feature space but hold different samples [14], as with two regional banks recording the same customer features for different clienteles. Each adds samples while the architecture and feature set stay uniform. HFL is the most studied category and the setting assumed by most optimisation techniques, including FedAvg [1], Scaffold [2], and FedProx [4]; the experiments in this study are horizontal.

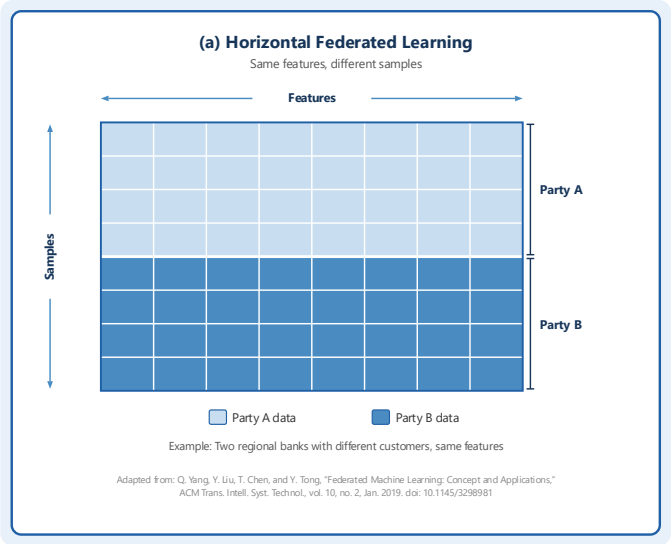


Figure 2.2: HFL Category of Federated Learning by Data Partitioning. (Source: Adapted from Yang et al. [14]).

ii. Vertical Federated Learning (VFL), or feature-based FL, applies when clients share the same samples but hold different features [14], as when a bank and an e-commerce firm serve overlapping customers with complementary financial and behavioural data. VFL lets them train jointly without disclosing data, but is more complex than HFL, requiring entity alignment and typically split-model architectures across parties [14], [15].

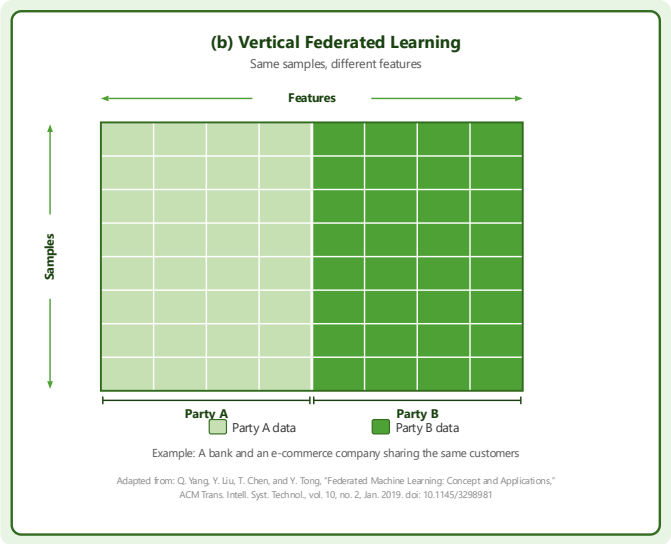


Figure 2.3: VFL Category of Federated Learning by Data Partitioning. (Source: Adapted from Yang et al. [14]).

iii. Federated Transfer Learning (FTL) handles clients differing in both samples and features, where neither HFL nor VFL applies [14]. It combines FL with transfer learning to share knowledge despite minimal overlap, widening FL's applicability but remaining the least explored category, with further challenges in domain adaptation and model alignment [14]. Figures 2.2 through 2.4 illustrate the three categories.

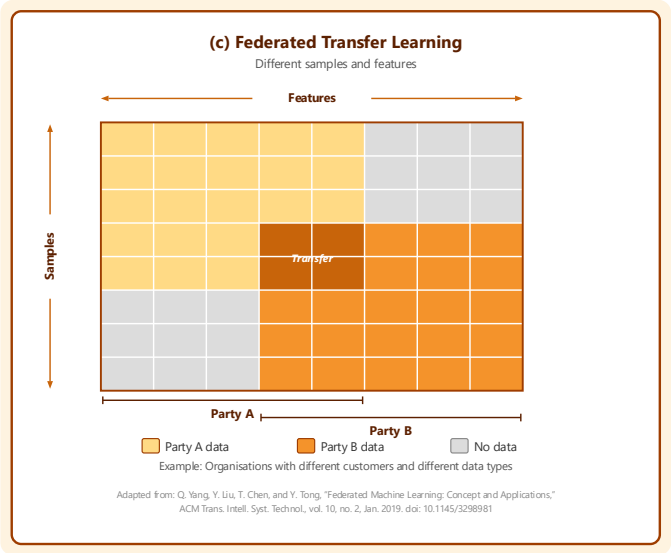


Figure 2.4: FTL Category of Federated Learning by Data Partitioning. (Source: Adapted from Yang et al. [14]).

2.2.4 Categories of Federated Learning by Scale Setting

Independent of the data-partitioning taxonomy, [15] distinguish two settings by the scale of the participating clients: cross-device and cross-silo:

i. Cross-device FL spans potentially millions or billions of devices such as smartphones and IoT sensors [15]: each has limited compute, may go offline unpredictably, and uses bandwidth-constrained networks; only a tiny fraction participate per round, and clients are stateless. This setting motivated FedAvg [1], with Google's Gboard next-word prediction a flagship deployment [15].

ii. Cross-silo FL involves a smaller set of organisational clients, typically two to a hundred, such as hospitals in an imaging consortium, banks building credit-risk models, or telecoms detecting network anomalies [15]. These clients are usually always available, well-resourced, on reliable high-bandwidth links, and stateful, retaining and updating information such as Scaffold's control variates [2] across rounds. This study's setup, 100 clients sampled at 20%, is cross-silo.

2.2.5 Key Challenges in Federated Learning

Federated learning introduces fundamental challenges that arise little or not at all in centralised machine learning:

i. Client drift. When clients run multiple local updates on non-IID data, their models drift toward local optima that may diverge sharply from the global optimum [2], [3]. This client drift, the central problem of this study, slows convergence, inflates the rounds needed for target accuracy, and can prevent convergence entirely; its severity grows with data heterogeneity and the number of local steps per round [2].
ii. Communication efficiency. In many deployments the cost of transmitting updates dominates training [1], [16]. For models with millions of parameters each round moves substantial data, so reducing rounds or compressing updates (via gradient compression or quantisation) are active research areas [14].
iii. Privacy and security. Although raw data is not shared, transmitted updates can still leak sensitive information: Zhu et al. [18] reconstructed training images from shared gradients, a vulnerability termed gradient leakage. Defences such as differential privacy, secure aggregation, or homomorphic encryption each trade off privacy, utility, and cost [15], [19].
iv. Fairness and heterogeneity. Because clients contribute unequally in data quantity and quality, a globally averaged model may perform well on average yet poorly for under-represented clients [15]. Ensuring the model serves all participants, not just aggregate performance, remains an open challenge.

2.2.6 Applications of Federated Learning

FL has been applied across domains where privacy, regulation, or logistics prevent data centralisation:

i. Healthcare. FL lets hospitals and research institutions jointly train diagnostic models, such as for medical-image classification, without sharing patient data [20], a key benefit under regimes such as HIPAA. Rieke et al. [20] identify it as an enabler of next-generation digital health, where large multi-institutional datasets are needed but sharing is legally or ethically constrained.
ii. Mobile computing and NLP. FL has been deployed at scale for next-word prediction, query suggestion, and voice recognition on mobile devices [1], [15]; Google's Gboard, training across millions of devices on text that never leaves the phone, is the foremost cross-device example.

iii. Financial services. Institutions are exploring FL for credit scoring, fraud detection, and anti-money-laundering, where pooling bank data would help but is barred by competition and regulation [14], [18].

iv. Autonomous driving and IoT. FL lets vehicles and edge devices contribute to shared perception and decision models without transmitting the vast sensor data they generate [18], and lets large IoT networks train anomaly-detection or predictive-maintenance models locally.

These applications underscore the value of improving the convergence and efficiency of federated optimisation, the central concern of this study.

2.3 Client Drift in Federated Learning

2.3.1 The Non-IID Data Problem

The performance of FL depends critically on how data is distributed across clients. Ideally the data would be independently and identically distributed (IID), so that each client's dataset is a representative sample of one underlying distribution; each local gradient is then an unbiased estimate of the global gradient, and averaging local updates approximates centralised training [1], [16]. In practice this rarely holds: each client's data reflects its own activities, users, or population, so a cardiology hospital differs sharply from an oncology one, and a French-typing phone user from a Mandarin-typing one, and local distributions can differ substantially in their labels, features, and sample sizes [3], [16], [21].

Zhao et al. [3] gave the first systematic empirical study, showing FedAvg accuracy can fall by up to 55% under highly non-IID data: with each client holding only one of ten classes (extreme label skew), accuracy dropped 51% on CIFAR-10 and 55% on keyword spotting. They attributed this to weight divergence, where parameters learned on skewed local data pull in different directions so their average is a poor compromise, and quantified it via the earth mover's distance (EMD) between each client's class distribution and the population's, finding degradation rose with EMD [3].

Zhu et al. [21] surveyed non-IID FL across parametric and non-parametric models in horizontal and vertical settings, categorising five forms of skew: label distribution skew (differing class proportions), feature distribution skew (a label presenting

differently), quantity skew (differing data volumes), temporal skew (distributions changing over time), and quality skew (differing noise or annotation accuracy) [21]. Label distribution skew is the most studied and is the form simulated here using the Hsu-Qi-Brown partitioning scheme. Figures 2.5 and 2.6 contrast IID and non-IID distributions across clients.

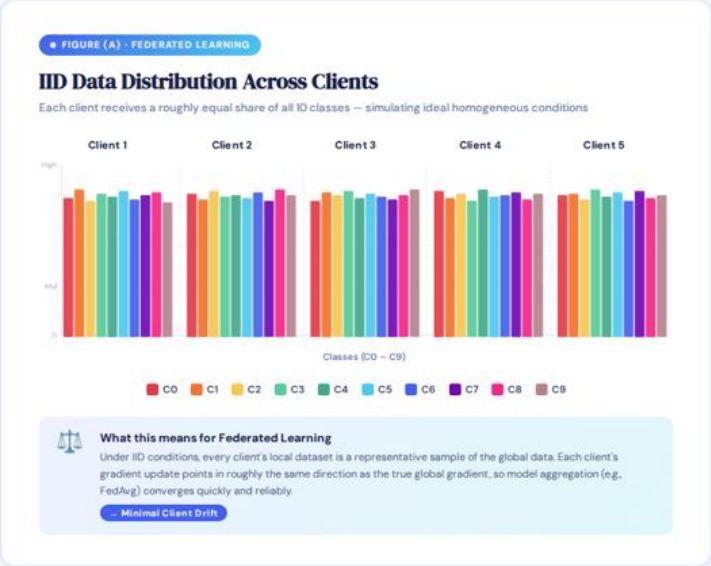


Figure 2.5: IID vs Non-IID Data Distribution Across Clients in Federated Learning. (Source: Adapted from Zhao, et al. [3]).

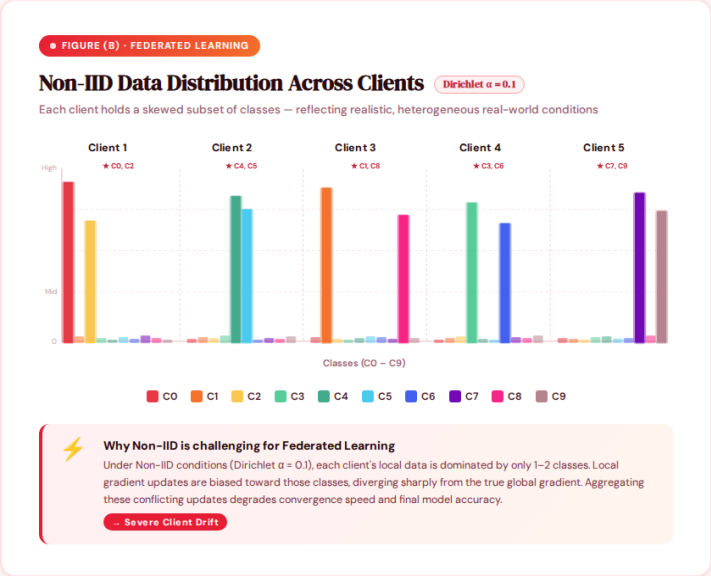


Figure 2.6: IID vs Non-IID Data Distribution Across Clients in Federated Learning. (Source: Adapted from Zhao, et al. [3]).

Li et al. [22] complemented these findings with a convergence analysis of FedAvg on non-IID data, proving an O(1/T) rate for strongly convex, smooth problems but showing that heterogeneity directly slows convergence. Crucially, without learning-rate decay FedAvg converges to a point $\Omega(\eta)$ from the true optimum, so it is fundamentally biased under non-IID data, not merely slowed. The number of local epochs E acts as a control knob: too few wastes local computation, while too many amplifies drift and degrades the final model [22].

Wang et al. [23] framed the same problem as objective inconsistency: in FedAvg each client minimises its own local loss, which under heterogeneous data differs from the global loss, so aggregating locally optimised models does not minimise the global objective. This mismatch is the root cause of client drift [23].

2.3.2 Understanding Client Drift

Client drift occurs when clients perform local gradient updates on their own non-IID data between rounds, so each client's parameters move toward a local optimum specific to its distribution rather than the global optimum [2], [15]. The term captures the directional divergence between where local training leads a client and where the global objective requires it to go.

For instance, across five hospitals training a shared chest-X-ray classifier, one seeing mostly pneumonia, another mostly healthy patients, and a third specialising in tuberculosis, several local epochs make each model excel on its own caseload while forgetting others. Averaging these divergent models yields a global compromise good for none: each client drifts toward its own specialisation.

Karimireddy et al. [2] formalised this in Scaffold, proving tight FedAvg convergence bounds that expose client drift as the principal bottleneck in heterogeneous settings. The rate depends on a gradient dissimilarity parameter G measuring how far local gradients diverge from the global one; when G is large (high heterogeneity), FedAvg is provably slower than centralised SGD, and the gap widens with more local steps. This is not merely insufficient communication: even with frequent aggregation, accumulated directional error from local steps introduces a systematic bias FedAvg

cannot correct [2]. Figures 2.7 and 2.8 illustrate client drift under IID and non-IID conditions.

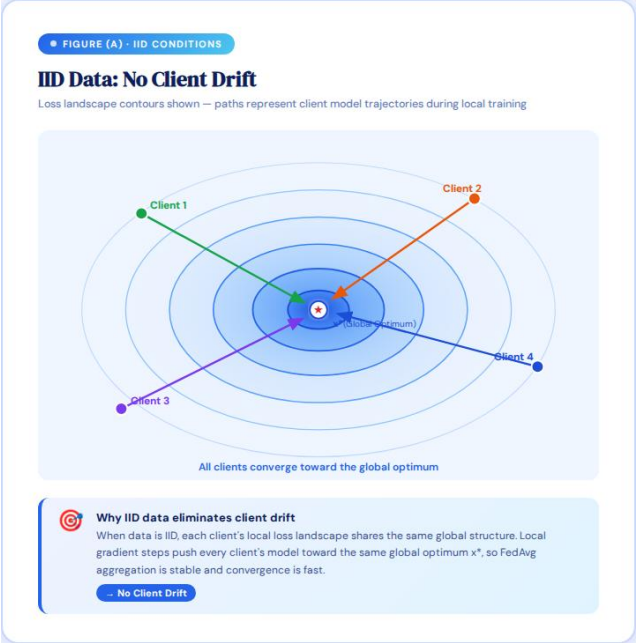


Figure 2.7: Client Drift Under IID (a) and Non-IID (b) Data Conditions



Figure 2.8: Client Drift Under IID (a) and Non-IID (b) Data Conditions

Drift severity is governed by two factors: the degree of statistical heterogeneity and the number of local steps between rounds [2], [22]. Greater heterogeneity makes local gradients point in more divergent directions, and more local steps let divergence

accumulate before the server can correct it. This creates a core design tension: more local computation cuts communication (fewer rounds) but amplifies drift and degrades the aggregated model [1], [2]. Balancing these, or devising methods that take many local steps without drifting, is a central challenge in federated optimisation.

Li et al. [4] showed empirically, in FedProx, that systems heterogeneity worsens drift: faster clients completing more local epochs drift further than slower ones, adding imbalance to the aggregated updates. Client drift is thus a joint data-and-systems problem any practical deployment must address [4].

BN-Scaffold [11] showed drift extends beyond the gradient-level divergence [2] characterised. Quintana et al. [11] found that batch-normalisation layers, which track running activation statistics, introduce a further drift when those statistics are computed on heterogeneous local data, and that standard Scaffold's control-variate correction does not address it, pointing to a need for more adaptive correction across the training process [11].

2.4 Theoretical Review

2.4.1 Optimization Theory in Distributed Systems

Federated learning is an instance of distributed stochastic optimisation. The starting point is empirical risk minimisation (ERM): given n examples and a model parameterised by w ∈ R^d, a single learner minimises the empirical risk:

$$\min \quad F(w) = (1/n) \sum_{i=1}^{n} \ell(w; \xi_i) \qquad \text{eq.1}$$

where $\ell(w; \xi_i)$ is the loss on example ξ_i. Centralised training solves this by stochastic gradient descent (SGD), with foundations in the stochastic-approximation work of [24]. At iteration t, SGD samples a mini-batch B_t of size b uniformly and updates:

$$w_{t+1} = w_t - \eta \cdot g_t, \text{ where } g_t = (1/b) \sum_{i \in B_t} \nabla \ell(w_t; \xi_i) \qquad \text{eq.2}$$

where $\eta > 0$ is the learning rate and g_t is an unbiased estimator of the true gradient ($E[g_t \mid w_t] = \nabla F(w_t)$) but carries sampling variance. The assumptions under which convergence guarantees follow are set out in [25]; two recur throughout the FL literature.

Assumption 1 (β-smoothness): A differentiable function F: R^d → R is β-smooth if its gradient is Lipschitz continuous with constant β > 0:

$$\|\nabla F(x) - \nabla F(y)\| \leq \beta \cdot \|x - y\|, \forall\, x, y \in R^d \qquad \text{eq.3}$$

Equivalently, $F(y) \leq F(x) + \langle \nabla F(x), y - x \rangle + (\beta/2)\|y - x\|^2$ [25]; smoothness is standard in the FedAvg and Scaffold analysis of [2].

Assumption 2 (μ-strong convexity):A differentiable function F is μ-strongly convex with parameter $\mu \geq 0$ if:

$$F(y) \geq F(x) + \langle \nabla F(x), y - x \rangle + (\mu/2)\|y - x\|^2, \forall\, x, y \in R^d \quad \text{eq.4}$$

With μ = 0 this is standard convexity. The ratio κ = β/μ is the condition number, governing first-order convergence speed [25]. Under Assumptions 1 and 2 with μ > 0, decaying-rate SGD achieves an O(1/T) rate on $E[F(w_t) - F(w^*)]$, with w* the global minimiser [24]-[25].

The distributed setting generalises ERM to a dataset partitioned across N workers. With each holding a local dataset D_i and the global objective a weighted sum of local empirical risks, the problem becomes:

$$\min \quad F(w) = \sum_{i=1}^{N} p_i \cdot F_i(w), \; F_i(w) = (1/|D_i|) \sum_{\xi \in D_i} \ell(w; \xi) \qquad \text{eq.5}$$

where $p_i \geq 0$ are weights summing to one, canonically $p_i = |D_i| / \sum_j |D_j|$, so each client contributes in proportion to its data [1], [15]. Two difficulties arise: workers see only local gradients $\nabla F_i(w)$, not $\nabla F(w)$; and communication is costly, motivating multiple local steps per round. Lower bounds in [15] prove the rounds R to reach ε-suboptimality cannot fall below $\Omega(\sqrt{(\kappa)} \cdot \log(1/\varepsilon))$ deterministically, regardless of algorithm, the benchmark for federated methods, which Scaffold matches up to acceleration [2].

A second strand is variance reduction. Classical SGD suffers gradient variance $\sigma^2 = E[\|\nabla \ell(w; \xi) - \nabla F(w)\|^2]$ that prevents exact convergence at a constant learning rate. SVRG [26] builds a variance-reduced estimator using control variates:

$$\tilde{g}_t = \nabla \ell(w_t; \xi) - \nabla \ell(\tilde{w}; \xi) + \nabla F(\tilde{w}) \quad \text{eq.6}$$

where $\tilde{w}$ is a periodically updated reference point for the full gradient. The estimator stays unbiased but its variance decays to zero as iterates approach $\tilde{w}$, enabling a constant learning rate and linear convergence for μ-strongly convex problems [26]; SAGA [27] achieves the same by storing per-sample gradients instead. This control-variate principle, borrowed from Monte Carlo estimation, reduces variance using a correlated quantity of known mean; applied to inter-client heterogeneity rather than mini-batch sampling, it underlies Scaffold (Subsection 2.4.3) [26].

2.4.2 Federated Learning (FL) and FedAvg

Building on Subsection 2.4.1, federated learning adds three constraints: data locality, partial participation, and statistical heterogeneity [1], [15]. The objective keeps the structure of Equation 5, but the datasets D_i come from distinct distributions (typically $D_i \neq D_j$) and raw data is never exchanged. FedAvg [1] is the foundational algorithm and the template for later techniques, including Scaffold.

FedAvg proceeds over R rounds. At round r the server holds a global model w^r and selects a subset $S^r \subseteq \{1, \ldots, N\}$ (typically uniform sampling without replacement, $|S^r| = S$), broadcasting w^r to each $i \in S^r$. Each client initialises $y_{i,0} = w^r$ and runs K steps of local SGD:

$$y_{i,k+1} = y_{i,k} - \eta_l \cdot g_i(y_{i,k}),\ k = 0, 1, \ldots, K-1 \quad \text{eq.7}$$

where η_l is the local learning rate and $g_i(y_{i,k})$ an unbiased stochastic estimate of $\nabla F_i(y_{i,k})$ with variance bounded by σ^2. After K steps each client sends the update $\Delta_i = y_{i,K} - w^r$, and the server aggregates:

$$w^{(r+1)} = w^r + \eta_m \cdot (1 / |S^r|) \sum_{i \in S^r} \Delta_i \quad \text{eq.8}$$

where $\eta_m \geq 1$ is the server learning rate. The original formulation [1] set $\eta_m = 1$, making the update a weighted average of client models. Decoupling the two rates tightens convergence [2]: a larger η_m permits a smaller η_l while still progressing, attenuating drift from the K local steps. The main obstacle in analysing FedAvg under heterogeneous data is the bias from local steps on objectives F_i that differ from F, formalised in [2] via:

Assumption 3 ((G, B)-Bounded Gradient Dissimilarity): There exist constants $G \geq 0$ and $B \geq 1$ such that for all $w \in R^d$:

$$(1/N)\sum_{i=1}^{N}\|\nabla F_i(w)\|^2 \le G^2 + B^2\cdot\|\nabla F(w)\|^2 \quad \text{eq.9}$$

Here G measures the residual disagreement between local gradients at global stationary points ($\nabla F(w) = 0$), and B the relative scaling of local- to global-gradient magnitudes. In the IID case G = 0 and B = 1; under non-IID data G > 0 and B > 1, both growing with heterogeneity [2]. Under Assumptions 1-3, the rate proved in [2] for μ-strongly convex objectives needs, to reach $E[F(w^R) - F(w^*)] \le \varepsilon$, rounds R satisfying:

$$R = \tilde{O}(\sigma^2/(\mu K S \varepsilon) + (1 - S/N) G^2/(\mu S \varepsilon) + \sqrt{\beta} G/(\mu\sqrt{\varepsilon}) + B^2\beta/\mu) \quad \text{eq.10}$$

where Õ suppresses logarithmic factors [2]. The $G/\sqrt{\varepsilon}$ and $B^2\beta/\mu$ terms make round complexity deteriorate with gradient dissimilarity, and a matching $\Omega(G^2/(\mu R^2))$ lower bound in [2] shows this is unavoidable even with full participation (S = N) and noiseless gradients ($\sigma = 0$). Centralised SGD achieves $O(\beta/\mu)$ complexity independent of G, so FedAvg is strictly worse under heterogeneous data. This is the formal statement of the client drift discussed in Subsection 2.3.2, motivating the variance-reduced Scaffold examined next.

2.4.3 Stochastic Controlled Averaging (Scaffold)

Scaffold (Stochastic Controlled Averaging for Federated Learning), proposed in [2], reduces FedAvg's gradient-dissimilarity dependence by applying the control-variate principle of Subsection 2.4.1 to inter-client heterogeneity. This correction is asymptotically unbiased and removes the gradient-dissimilarity terms from the convergence bound; in finite rounds, drift is controlled rather than eliminated, the residual bias being bounded by the control-variate approximation quality [2]. The divergence between local and global gradient directions is treated as structured variance, and the Monte Carlo correction behind SVRG [26] repurposed to correct it.

Scaffold maintains two extra state variables: a server control variate $c \in R^d$ estimating the average global gradient direction, and a per-client variate $c_i \in R^d$ for each i estimating its local direction. Both start at zero, and the invariant $c = (1/N)\sum_i c_i$ holds throughout [2]. At round r the server broadcasts w^r and c^r to each $i \in S^r$; each client initialises $y_{i,0} = w^r$ and runs K corrected local steps:

$$y_{i,k+1} = y_{i,k} - \eta_l \cdot (g_i(y_{i,k}) - c_i^r + c^r) \quad \text{eq.11}$$

The correction term ($c^r - c_i^r$) replaces the client-specific bias $g_i(y_{i,k}) - \nabla F_i(y_{i,k})$ with an estimate of the global direction, mirroring the SVRG estimator (Equation 6) with c_i^r in place of $\nabla \ell(\tilde{w}; \xi)$ and c^r in place of $\nabla F(\tilde{w})$. After the K steps, each client updates its control variate using one of two options [2]:

$$\text{Option I: } c_i^{r+1} = g_i(w^r); \text{ Option II: } c_i^{r+1} = c_i^r - c^r + (1 / (K \eta_l))(w^r - y_{i,K}) \quad \text{eq.12}$$

Option I needs an extra gradient evaluation at w^r and is more stable; Option II re-uses local-training gradients, cheaper but slightly less stable [2]. Each client transmits both $\Delta y_i = y_{i,K} - w^r$ and $\Delta c_i = c_i^{r+1} - c_i^r$, and the server aggregates:

$$w^{r+1} = w^r + (\eta_m / |S^r|) \sum_{i \in S^r} \Delta y_i, \; c^{r+1} = c^r + (1/N) \sum_{i \in S^r} \Delta c_i \quad \text{eq.13}$$

The server variate is averaged over N, not $|S^r|$, preserving the invariant under partial participation since non-participating clients keep their variates [2]. Two properties follow. First, holding all c_i and c at zero reduces Equation 11 to the FedAvg update of Equation 7, so Scaffold strictly generalises FedAvg. Second, it is stateful, each client retains c_i across rounds, distinguishing it from stateless methods such as FedProx [4] and suiting cross-silo settings with persistent participation [15].

The central result of [2] is that Scaffold's convergence rate is independent of the dissimilarity parameters G and B of Assumption 3. Under Assumptions 1 and 2 with $\mu > 0$, it attains:

$$R = \tilde{O}(\sigma^2 / (\mu K S \varepsilon) + \beta / \mu + N / S) \quad \text{eq.14}$$

where σ^2 is the per-client stochastic-gradient variance defined in Equation 2, K the number of local steps per round (Equation 7), $S = |S^r|$ the number of clients sampled each round, N the total number of clients, β and μ the smoothness and strong-convexity constants of Assumptions 1 and 2 respectively, and ε the target optimisation error communication rounds to reach ε-suboptimality, free of G and B [2]. Setting S = N recovers centralised SGD's rate with K-times larger batches, so Scaffold is no slower than SGD even on heterogeneous data. Under an additional δ-bounded Hessian-dissimilarity assumption, for quadratic objectives it attains $\tilde{O}(\sigma^2/(\mu K N \varepsilon) + 1/(\mu K)$

+ δ/μ), which improves monotonically with local steps K when δ is small, the first proof that local steps yield a genuine speedup beyond communication savings [2]. The proof bounds the bias of the corrected update via:

$$\sum_i \| (g_i(y) - c_i + c) - \nabla F(y) \|^2 \leq \sum_i \| c_i - \nabla F_i(y) \|^2 \quad \text{eq.15}$$

Here y denotes an arbitrary local iterate $y_{i,k}$ produced during a client's K corrected local steps (Equation 11); the bound therefore applies throughout local training, not solely at the communication-round boundary so the corrected gradient's deviation from the true global gradient depends only on the approximation quality $c_i \approx \nabla F_i(y)$, not on the dissimilarity between the F_i; under smoothness, ∇F_i varies slowly, keeping the variates accurate [2]. Despite these guarantees, Scaffold applies the same local rate η_l uniformly at every step, never adapting to the alignment between a client's corrected direction and the global trajectory, motivating the cosine-similarity modulation grounded in the alignment theory developed next.

2.4.4 Cosine Similarity Theory

Cosine similarity measures the directional alignment between two non-zero vectors in an inner-product space. It originated in the vector-space model of information retrieval [28] for ranking documents against query vectors; the canonical treatment in [29] situates it among vector-based similarity measures in machine learning. This subsection gives its definition, the properties relevant to gradient-direction analysis, and the basis on which CosScaffold appropriates it for adaptive drift modulation.

Definition (Cosine Similarity). For two non-zero vectors u, v ∈ R^d, the cosine similarity is:

$$\cos(u, v) = \langle u, v \rangle / (\|u\| \cdot \|v\|) = (\sum_{i=1}^{g} u_i v_i) / (\sqrt{(\sum_i u_i^2)} \cdot \sqrt{(\sum_i v_i^2)}) \quad \text{eq.16}$$

where $\langle u, v \rangle$ is the Euclidean inner product and $\|u\| = \sqrt{\langle u, u \rangle}$ the induced ℓ^2 norm. It is undefined when either vector is zero, handled in implementations by identity clamping, a property the CosScaffold modulator exploits. Because numerator and denominator share the same homogeneity degree, it is invariant under positive scaling of either argument: $\cos(\alpha u, v) = \cos(u, \alpha v) = \cos(u, v)$ for $\alpha > 0$ [28]-[29]. This scale invariance distinguishes it from inner-product or Euclidean measures and is the main

reason for its use here: a client's gradient magnitude depends on local data size and step count, whereas a directional measure should not.

Cosine similarity lies in $[-1, +1]$, a direct consequence of the Cauchy-Schwarz inequality $|\langle u, v\rangle| \leq \|u\| \cdot \|v\|$ [28]: $+1$ denotes perfect alignment ($u = \alpha v$, $\alpha > 0$), 0 orthogonality, and -1 directly opposed vectors ($\alpha < 0$). Its relation to the angle θ between u and v is:

$$\cos(u, v) = \cos(\theta), \theta \in [0, \pi] \quad \text{eq.17}$$

which justifies the name. Expanding $\|u - v\|^2 = \|u\|^2 + \|v\|^2 - 2\langle u, v\rangle$ and rearranging links it to squared Euclidean distance:

$$\|u - v\|^2 = \|u\|^2 + \|v\|^2 - 2 \cdot \|u\| \cdot \|v\| \cdot \cos(u, v) \quad \text{eq.18}$$

so cosine similarity captures the angular component of inter-vector distance, decoupled from magnitude. In FL, two client updates Δy_i and Δy_j may have comparable ℓ^2 norms (set by η_l and K) yet very different angles, and averaging in Equations 8 and 13 cancels that angular discrepancy. Cosine similarity is thus the natural measure of destructive interference between updates.

Cosine similarity is not a strict metric: though symmetric and bounded, the derived distance $d(u, v) = 1 - \cos(u, v)$ generally violates the triangle inequality [28]. This is inconsequential here, where it serves as an alignment score, not a metric. Computationally it costs O(d), one inner product and two norms, so for $d > 10^6$ it is negligible beside a single stochastic gradient, justifying its use as a real-time signal during local training in CosScaffold [7]-[8]. Cosine similarity has served related but distinct purposes in FL. In [7], pairwise cosine similarity between client updates forms aggregation weights that down-weight clients pointing away from the majority. In [8], it underpins a Byzantine-robust aggregation rule excluding malicious clients whose updates are anomalously misaligned. In [9], it supports a gradient-based defence against data-leakage attacks in vertical FL by flagging directional deviations. In [30], it is the similarity kernel in the MOON contrastive-learning framework, applied to feature representations rather than gradients. None, however, treats it as a real-time modulator of the local learning rate during client-side training, the specific contribution of CosScaffold, developed in Chapter Three on the foundations of Subsections 2.4.1 to 2.4.4.

2.5 Existing Techniques for Client Drift Mitigation

2.5.1 Aggregation-Based Methods

The simplest drift-mitigation category acts at aggregation, changing how the server combines updates, not local training. FedAvg [1] (Section 2.2) averages client parameters by dataset size but cannot correct directional divergence under non-IID data.

In [23], a distinct FedAvg failure is identified: when clients take different numbers of local steps, the weighted average converges to a mismatched surrogate objective, not the true one. The proposed FedNova normalises each client's accumulated update by its step count before aggregation, removing this inconsistency; its framework subsumes FedAvg and FedProx, showing the proximal term mitigates but does not eliminate this bias. FedNova corrects only aggregation, leaving the gradient-level drift from local training untouched, complementing rather than replacing trajectory-correcting methods [23].

FedMA [31] tackles a structural flaw: because networks are permutation-invariant, equivalent neurons may occupy different positions across models. It aligns neurons before averaging via layer-wise Bayesian nonparametric matching, beating FedAvg and FedProx on VGG-9 and LSTM. Its limits are practical: architecture-specific, costly for wide networks, and not easily extended to transformers or residual networks [31].

2.5.2 Regularization-Based Methods

Regularisation methods add penalties to the local objective that discourage deviation from the global model; simple to implement, they address the magnitude of drift, not its direction.

FedProx [4] adds a proximal term $\mu/2 \, \|w - w^t\|^2$ penalising the squared distance between local and global parameters, and introduces γ-inexact solutions for stragglers. It improves test accuracy by an average of 22% over FedAvg in highly heterogeneous settings and guarantees $O(1/T)$ convergence under a B-local dissimilarity assumption [4]. The penalty is isotropic, however, constraining all directions equally and treating

harmful drift and useful specialisation alike; as shown in [2], it offers no formal convergence advantage over FedAvg, both rates degrading identically with G.

MOON [30] applies contrastive learning at the representation level: a local contrastive loss keeps the current model's features close to the global model's (positive pair) and away from the previous local model's (negative pair), via cosine similarity under temperature τ. It outperforms FedAvg, FedProx, and Scaffold on CIFAR-10 and CIFAR-100 [30], but requires storing a previous local model per client, tuning several hyperparameters, and is validated mainly on image classification.
The common thread is that regularisation keeps updates near the global model without correcting the gradient-level bias behind drift, treating a symptom (parameter divergence) rather than the root cause (directional divergence of local from global gradients).

2.5.3 Variance Reduction via Control Variates

Scaffold [2] is the most theoretically principled drift-mitigation approach. Borrowing the variance-reduction framework of centralised optimisation (SVRG, SAGA), it adds control variates that estimate and correct the gradient-level discrepancy between local and global objectives. The server holds a global variate c approximating the global gradient direction and each client i a local c_i for its own; the local step changes from $y_i \leftarrow y_i - \eta_l \cdot g_i(y_i)$ to $y_i \leftarrow y_i - \eta_l \cdot (g_i(y_i) + c - c_i)$, where $(c - c_i)$ steers the update back toward the global objective [2]. Each client then transmits its model update and a control-variate delta.

Scaffold's central result is a convergence rate independent of the gradient-dissimilarity parameter G. Heterogeneity is formalised in [2] as (G, B)-Bounded Gradient Dissimilarity, G capturing gradient disagreement at a stationary point and B the relative scaling; the proven strongly convex rate has no G or B terms, so communication complexity matches centralised SGD however heterogeneous the data. FedAvg's rate instead carries terms proportional to G^2/ε, and a matching lower bound shows this is unavoidable [2]. For quadratic objectives, local steps give a genuine speedup beyond communication savings, the first formal quantification of this.

This G-independence sets Scaffold apart from every technique here; no regularisation method (FedProx, MOON), parameter-level correction (FedDC), or aggregation adjustment (FedNova) achieves it. Scaffold has three limitations. First, transmitting control-variate deltas doubles the per-round communication cost versus FedAvg [2]. Second, and most relevant here, the local learning rate is uniform across all clients and steps: a well-aligned client takes the same step as a severely drifting one, over-perturbing when alignment is strong and under-restraining when weak. Third, results have been inconsistent, in [32], Scaffold's accuracy was notably unstable, and under extreme non-IID conditions (one class per client) FedProx outperformed it. These limitations, especially the uniform local rate, motivate the adaptive extension developed here.

2.5.4 Dynamic Regularization

FedDyn [5] bridges variance reduction and regularisation. Each client solves a modified local objective with a dual variable h_i as a dynamic penalty that, via an ADMM-inspired primal-dual view, aligns the local optimality conditions with the global objective. Like Scaffold, it converges independently of heterogeneity, but with better partial-participation dependence and only standard per-round communication, the correction being server-side [5]. Its scalability is limited, however. In [10], h_i grows unbounded over rounds: as its norm rises, test loss increases and accuracy degrades, an instability even when FedDyn's assumptions hold. The server must also store state proportional to client count times model dimension, prohibitive for cross-device FL [10].

2.5.5 Parameter-Level Drift Correction

FedDC [6] introduces auxiliary drift variables tracking the cumulative gap between local and global parameters; its local objective adds a constraint penalty and a gradient-correction term, updating the drift variable each round. It needs only standard communication (as FedAvg) and outperforms Scaffold on CIFAR-10 and CIFAR-100 [6]. The distinction from Scaffold is the level of correction: Scaffold corrects each local SGD step at the gradient level, whereas FedDC corrects accumulated parameter-level deviation. FedDC thus addresses a consequence of drift (parameter divergence) rather than its root cause (directional gradient divergence), so it does not attain

Scaffold's G-independence and is theoretically weaker despite strong empirical results [6].

2.5.6 Personalization-Based Methods

Personalisation methods accept that heterogeneous clients may need different models, treating the global model as an initialisation or regularisation anchor for each client's own.

In [33], Per-FedAvg recasts FL as MAML-based meta-learning, optimising a shared initialisation so that a few local gradient steps yield a good personalised model; it guarantees convergence for non-convex losses but needs expensive Hessian-vector products and only single-step adaptation [33]. In [34], this is improved via bi-level optimisation with Moreau envelopes: each client minimises its local loss plus a proximal penalty tying it to the global model, updated by standard averaging. Needing only first-order gradients, it achieves a quadratic convergence speedup for strongly convex objectives [34]. Ditto [35] trains a personalised model v_k per client alongside the global model, regularised toward it, reducing cross-device variance while improving accuracy and adding robustness to poisoning, since a corrupted global model lets personalised models deviate from it [35].

Personalisation suits clients with enough local data under severe heterogeneity, but trades global quality for per-client performance and does not improve the shared global model, standard FL's primary objective [16].

2.5.7 Knowledge Distillation Approaches

Knowledge distillation (KD) handles heterogeneity by transferring knowledge through soft predictions rather than parameter averaging, also allowing heterogeneous client architectures.

FedDF [36] replaces parameter averaging with server-side ensemble distillation on a public or unlabelled dataset, minimising the KL divergence between the global model's predictions and clients' averaged soft predictions. It supports heterogeneous architectures and needs far fewer rounds than FedAvg [36], but requires a public dataset at the server, often infeasible in privacy-sensitive domains. FedNTD [37] takes

a continual-learning angle: a Not-True Distillation loss preserves the global model's knowledge on non-true classes via KL divergence on not-true class logits only, without disturbing local classification. It needs no public dataset and no extra communication, modifying only the local objective [37], but works at the output-logit level rather than correcting the gradient-level drift causing the convergence difficulties of Section 2.3.

2.5.8 Comparative Summary

The techniques reviewed, summarised in Table 2.1 below, span aggregation modification, local-deviation constraints, gradient-direction correction, objective alignment, personalisation, and knowledge distillation. Each offers a valuable insight and carries limitations leaving room for improvement.

Scaffold is the most suitable basis for this study's extension. First, it is the only technique whose convergence rate is formally independent of the gradient-dissimilarity parameter G, the strongest guarantee against degradation as heterogeneity grows; no regularisation, parameter-level, or aggregation-based method achieves this. Second, it corrects drift's root cause at the gradient level (directional divergence of local gradients) rather than a downstream consequence. This makes each client's local step size the natural target for adaptive enhancement: modulating the effective learning rate by the real-time alignment between local and global directions extends the control-variate framework naturally.

The limitation this study addresses, Scaffold's uniform local learning rate, is an opportunity for refinement: a uniform step size across all clients and steps ignores how drift severity varies across clients and training stages. The proposed CosScaffold (Cosine Similarity extended Scaffold) introduces cosine similarity as an adaptive scaling mechanism for the local learning rate, preserving Scaffold's foundations while the step size responds to the drift at each step.

Across these four techniques, a consistent pattern emerges: methods that act at the parameter or objective level (FedProx, FedDyn, FedDC) offer practical, low-overhead improvements but do not correct drift at its root -the directional divergence of local from global gradients -and in FedDyn's case introduce their own instability through unbounded auxiliary state. Scaffold alone corrects drift at the gradient level with a

dissimilarity-independent guarantee, but applies that correction uniformly regardless of how aligned or misaligned a client's update already is. CosScaffold is positioned precisely at this remaining gap: it retains Scaffold's gradient-level correction and its convergence guarantee in full, while using a bounded, direction-sensitive cosine-similarity modulator -rather than an unbounded dual variable (FedDyn) or a parameter-level auxiliary term (FedDC) -to address the one limitation none of these methods resolves.

Table 2.1: Comparative Summary of Client Drift Techniques.

FL Approach / Paradigm	**Key Mechanism**	**Main Strengths**	**Primary Limitations**
Aggregation-based (FedNova, FedMA)	Corrects the server-side combination rule -normalising by local step count (FedNova) or matching neurons before averaging (FedMA) -without touching local training.	No change to client-side computation; FedNova subsumes FedAvg/FedProx as special cases; FedMA improves accuracy on non-IID VGG-9/LSTM benchmarks.	Leaves gradient-level drift from local training untouched; FedMA is architecture-specific and costly for wide networks or transformers.
Regularisation -based (FedProx, MOON)	Adds a penalty term to the local objective that discourages deviation from the global model (proximal term) or from the global representation (contrastive term).	Simple to implement; FedProx improves accuracy by ~22% over FedAvg on average under heterogeneity; MOON outperforms FedAvg/FedProx/Scaffold on CIFAR-10/100.	Penalty is isotropic (FedProx) - cannot distinguish harmful drift from useful specialisation; offers no formal convergence advantage over FedAvg; MOON needs a stored previous local model and careful temperature tuning.

Variance reduction via control variates (Scaffold)	Maintains a server and per-client control variate estimating gradient direction; corrects each local SGD step directly at the gradient level.	Convergence rate provably independent of gradient dissimilarity (G, B); strictly generalises FedAvg; benefits from more local steps rather than suffering from them.	Doubles per-round communication; applies the same local learning rate to every client regardless of alignment; empirically unstable under extreme non-IID partitions.
Dynamic regularisation (FedDyn)	Each client adds a dynamically updated linear (ADMM-style) penalty term, aligning local optimality conditions with the global objective.	Convergence independent of heterogeneity; only standard per-round communication (correction is server-side).	Internal dual-variable state can grow unbounded, causing asymptotic instability; requires prior knowledge of total client count and frequent resampling, limiting cross-device use.
Parameter-level drift correction (FedDC)	Tracks a cumulative gap between local and global parameters via auxiliary drift variables, correcting at the parameter level.	Standard (FedAvg-level) communication cost; strong empirical results, outperforming Scaffold on CIFAR-10/100.	Corrects a symptom (parameter divergence), not the root cause (gradient-direction divergence); does not attain Scaffold's dissimilarity-independent guarantee.

Personalisation-based (Per-FedAvg, pFedMe, Ditto)	Treats the global model as a shared initialisation or regularisation anchor and trains a personalised model per client.	Formal convergence guarantees (incl. quadratic speedup for pFedMe); Ditto adds robustness to poisoning.	Optimises for per-client performance, not shared global model quality -the primary FL objective; Per-FedAvg needs costly Hessian-vector products.
Knowledge distillation (FedDF, FedNTD)	Transfers knowledge via soft predictions/logits rather than parameter averaging (server-side ensemble distillation or client-side non-true-class KL).	Naturally supports heterogeneous client architectures; constrains decision boundaries more richly than parameter-level penalties.	Requires a public/unlabelled proxy dataset (FedDF); adds KL-divergence tuning overhead; validated mainly on classification tasks.
Feedback-control-based (FedPIDAvg)	Reframes aggregation as a PID controller - proportional, integral, and derivative feedback terms on local–global "error".	Offers an interpretable lens that also re-explains Scaffold (control variates ≈ feedback correction) and FedDyn (dual update ≈ integral term).	Adds three additional hyperparameters to tune; empirical validation is comparatively limited.
Meta-learning-based (Per-FedAvg / MAML)	Learns a shared initialisation that adapts to each client in a few local gradient steps, rather than correcting drift directly.	Naturally accommodates heterogeneity by design; FOMAML variant avoids second-order derivatives at little cost.	Accepts, rather than corrects, drift -does not improve the single shared global model; adaptation quality depends on step-size tuning.

Contrastive-learning-based (MOON)	Applies a cosine-similarity-based contrastive loss at the representation level to pull local features toward the global model and away from the client's own previous state.	Richer signal than Euclidean parameter distance; directly demonstrates cosine similarity's utility in FL, though at the representation rather than gradient level.	Needs a stored previous local model per client; adds a temperature hyperparameter; validated mainly on image classification.
Gradient normalisation / momentum-based (FedNova, FedAvgM, SlowMo, FedOpt)	Normalises client updates by step count or smooths aggregated updates with momentum/adaptive optimisers.	Directly targets systems-heterogeneity-induced drift (differing local step counts), a distinct source from statistical heterogeneity.	Complementary rather than a complete solution -does not address gradient-level statistical drift on its own.

2.6 Design Paradigms and Theoretical Foundations for Client Drift Mitigation

2.6.1 Variance Reduction and Control Variates

Scaffold's roots lie in variance reduction from centralised stochastic optimisation. In stochastic gradient descent (SGD), a mini-batch gradient is unbiased but high-variance, slowing convergence and ruling out large learning rates. Control variates, from Monte Carlo estimation, reduce this variance without bias: for a variable X with mean μ and a correlated Y with known mean ν, the estimator $\tilde{X} = X - Y + \nu$ preserves the mean ($E[\tilde{X}] = \mu$) while lowering variance when X and Y are positively correlated [26].

SVRG, introduced in [26], put this into practice: it snapshots a full gradient at a reference point and corrects each update by the per-sample gradient difference between that point and the current one. Because the points stay close, the gradients are highly correlated and variance falls toward zero, allowing a constant learning rate and linear convergence on smooth, strongly convex functions [26]. SAGA [27] and SARAH [38] later removed the periodic full-gradient pass,the former by storing per-sample gradients, the latter by recursive estimation. Scaffold's insight [2] was that in federated learning the dominant "variance" stems not from sampling but from

objective heterogeneity: local SGD on non-IID data biases each client's gradient relative to the global one, compounding over local steps. Scaffold maintains a server control variate c for the global direction and a per-client variate c_i, giving $y_i \leftarrow y_i - \eta_l(g_i(y_i) + c - c_i)$; the term $(c - c_i)$ mirrors SVRG's correction, replacing the client bias with the global direction [2]. This grounding gives Scaffold convergence guarantees independent of gradient dissimilarity.

2.6.2 Proximal Operator Theory

A second tradition penalises deviation from the global model through proximal operators. The proximal operator, introduced in [39] in 1962, is $\text{prox}_{\lambda f}(v) = \text{argmin}_x \{f(x) + (1/2\lambda)\|x - v\|^2\}$, trading off minimising f against proximity to v. Its Moreau envelope, $M_{\lambda f}(v) = \min_x \{f(x) + (1/2\lambda)\|x - v\|^2\}$, is a smooth, always-differentiable approximation of f [39]. The proximal point algorithm, formalised in [40], iterates $z^{k+1} = \text{prox}(z^k)$, the quadratic penalty keeping each subproblem well-conditioned, with a modern survey in [41]. FedProx [4] applies this: each client minimises a proximal surrogate of its loss, $h_k(w; w^t) = F_k(w) + (\mu/2)\|w - w^t\|^2$, where w^t is the global model and μ the constraint strength. It anchors parameters that stray too far, so each local step approximates a proximal point iteration [4]. pFedMe [34] goes further, using the Moreau envelope in a bi-level scheme that yields personalised client models while learning the global reference by averaging. As noted in Section 2.3, the penalty is isotropic: it treats all directions of deviation alike and cannot separate harmful drift from useful local specialisation [2].

2.6.3 ADMM and Primal-Dual Optimisation

The Alternating Direction Method of Multipliers (ADMM), surveyed in [42], underpins dynamic regularisation. ADMM forms the augmented Lagrangian,the objective plus a linear dual term and a quadratic penalty,then alternates between minimising over the primal variables and updating the dual variables. These dual variables accumulate constraint-violation history as persistent memory enforcing consensus: each agent's local variable must agree with the others, while the duals track and correct any deviation [42]. FedDyn [5] adds a linear term $-\langle h_k^t, \theta \rangle$ to the quadratic penalty, with the dual variable updating as $h_k^{t+1} = h_k^t - \alpha(\theta_k^{t+1} - \theta^t)$. It accumulates local–global disagreements, so persistent deviation in one direction grows a counteracting correction; at convergence $h_k^* = \nabla f_k(\theta^*)$, recovering global stationarity

when summed across clients [5]. Unlike proximal methods the correction is dynamic and directional rather than static and isotropic, though the dual variable can grow without bound and destabilise training [10].

2.6.4 Meta-Learning

Model-Agnostic Meta-Learning (MAML) [43] learns an initialisation that adapts to new tasks in a few gradient steps. Its bi-level objective tunes θ by evaluating post-adaptation performance, thus learning adaptability; the meta-gradient needs second-order derivatives, though first-order FOMAML performs almost as well [43]. Per-FedAvg [33] recasts federated learning as meta-learning: rather than one model for all clients, it learns a meta-initialisation each client personalises locally. The global model becomes a shared starting point, letting clients adapt differently to accommodate heterogeneity. Unlike correction-based methods, this approach accepts that clients need different solutions and simply seeks the best common start [33].

2.6.5 Contrastive Learning

Contrastive self-supervised learning, established by SimCLR [44], learns representations by drawing augmented views of an image together (positive pairs) and pushing different images apart (negative pairs). Its normalised temperature-scaled cross-entropy (NT-Xent) loss uses cosine similarity between projected representations, scaled by a temperature τ that sets the distribution's sharpness [44]. MOON [30] shifts this to model-level alignment: a contrastive loss pulls the local model toward the global representations and away from its previous state. Operating in representation rather than parameter space gives a richer signal than FedProx's Euclidean distance [30]. Its reliance on cosine similarity is notable: the measure aligns effectively in federated learning, though over representations rather than the gradient directions used in CosScaffold (Cosine Similarity extended Scaffold).

2.6.6 Knowledge Distillation

Knowledge distillation [45] showed that a teacher's soft, temperature-scaled softmax outputs carry rich inter-class information that one-hot labels discard; a student matching them learns class-similarity structure, not just the correct class. The loss combines cross-entropy with a KL divergence between softened teacher and student

outputs, plus a T^2 factor offsetting the temperature-induced gradient-magnitude reduction [45].

FedDF [36] distils on the server, treating client models as a teacher ensemble distilled into one global student via proxy data; acting on outputs, not parameters, it supports heterogeneous architectures [36]. FedNTD [37] distils on the client, taking the KL divergence only over non-true-class logits, so the model learns the true class locally while resisting drift elsewhere [37]. Distillation thus constrains output behaviour and decision boundaries,richer than parameter-level constraints, as soft labels encode structural class relationships.

2.6.7 Feedback Control Theory

Classical PID control combines three feedback terms: proportional (current error), integral (removing steady-state offset via accumulated past errors), and derivative (anticipating error from its rate of change). This maps onto the federated server-client cycle, where the "error" is local–global drift. FedPIDAvg [46] forms the global model as a PID-weighted average of client models, weighting by sample count (proportional), cost drop since the previous round (derivative), and cumulative cost over rounds (integral) [46]. The lens also reframes existing methods: Scaffold's control variates act as feedback corrections, FedDyn's dual updates as an integral controller.

2.6.8 Gradient Normalisation and Momentum

A final thread addresses heterogeneous local computation: when clients take differing local-step counts, FedAvg's averaging converges to a mismatched surrogate, not the true objective, which FedNova corrects by normalising each client's update by its effective step count [23]. Server-side momentum (FedAvgM, SlowMo [47]) smooths noisy aggregated updates with averaged gradient history, while FedOpt [48] decouples client and server optimisation so adaptive optimisers (FedAdam, FedYogi) normalise gradients via adaptive learning rates [48]. These complement variance reduction and proximal methods on a different facet of heterogeneity.

2.7 Results of the Systematic Literature Review (Objective 1)

Building on the individual techniques reviewed in Section 2.4 and the design paradigms examined in Section 2.5, this section reports the findings of the PRISMA 2020-compliant systematic literature review [12]–[13] carried out to satisfy Objective One: investigating the existing approaches, characteristics, and limitations relevant to extending Scaffold for client-drift mitigation in Federated Learning. The review identified the dominant strategies in the literature, set out the limitations that restrict their practical use, and exposed the gap in similarity-based mechanisms that motivated the CosScaffold (Cosine Similarity extended Scaffold) technique developed in Chapter Three.

2.7.1 SLR Search Strategy and Protocol

The review followed the PRISMA 2020 reporting framework [12]–[13], which gives a transparent, reproducible structure for identifying, screening, assessing, and including studies. Three research questions framed it: (i) which approaches have been used to extend Scaffold and control-variate techniques for client-drift mitigation in Federated Learning; (ii) what limitations they carry; and (iii) which characteristics of Scaffold are most often targeted for extension. These map directly onto the three sub-questions of Objective One.

The search combined three Boolean groups covering Scaffold-specific extensions and broader control-variate and variance-reduction work on client drift. The composed string was: (Scaffold OR Scaffold Enhancement OR Scaffold Optimization OR Scaffold Modification OR Scaffold Extension OR Control Variates OR Variance Reduction) AND (Client Drift OR Client Drift Mitigation OR Client Drift Correction OR Client Drift Reduction OR Data Heterogeneity OR Non-IID) AND (Federated Learning OR Federated Optimization). Seven databases were searched, IEEE Xplore, Science Direct, Springer, ACM Digital Library, Semantic Scholar, arXiv, and Google Scholar, together spanning the peer-reviewed literature, including the major machine-learning venues (NeurIPS, ICML, ICLR, AAAI). The 2016–2026 window captured both the original FedAvg work [1] and later extensions following the 2020 introduction of Scaffold [2]; only English-language publications were retained, with the final search run in February 2026.

Included studies were peer-reviewed articles, published proceedings, or established preprints within 2016–2026 that addressed Scaffold, its extensions, control variates, variance reduction, or client-drift mitigation in Federated Learning, and that reported at least two of the five performance metrics (convergence rate, communication efficiency, model accuracy, client-drift reduction, computational overhead). Non-peer-reviewed work without significant citations, non-English studies, duplicates, non-Federated-Learning work, and studies addressing fewer than two metrics were excluded. Risk of bias was limited by a quality threshold requiring each study to answer all three research questions, alongside the multi-database, comprehensive Boolean search above. The seven databases yielded 1,847 records. Removing 412 duplicates left 1,435 for title-and-abstract screening, of which 1,240 were excluded for not addressing the target topics or for reporting fewer than two metrics. Of 195 reports sought for full text, seven were unobtainable, leaving 188 for eligibility assessment; a further 155 were then excluded (47 not centred on Scaffold or control variates, 39 not primarily on client-drift mitigation, 28 addressing fewer than two metrics, 24 not on Federated Learning, 11 outside 2016–2026, and 6 non-English). After quality assessment, 33 studies were retained. Three denominators are used inline: n = 33 for the full corpus; n = 32 for the empirical sub-corpus (excluding one theoretical analysis with no proposed extension); and n = 30 for the characteristic-coding sub-corpus (further excluding two framework-level studies not framed at a specific Scaffold characteristic). Table 2.1 gives the distribution of included studies across the seven databases, and Fig. 2.9 presents the PRISMA 2020 flow diagram.

Table 2.2: Distribution of Included Studies across the Seven Digital Libraries.

Digital Library/Database	Identified (after search)	Selected as relevant	Included after QA
IEEE Xplore	285	38	10
Semantic Scholar	412	48	11
Google Scholar	489	42	6
arXiv	274	30	2
Science Direct	128	12	1
Springer	130	16	1
ACM Digital Library	129	9	2
Total	**1,847**	**195**	**33**

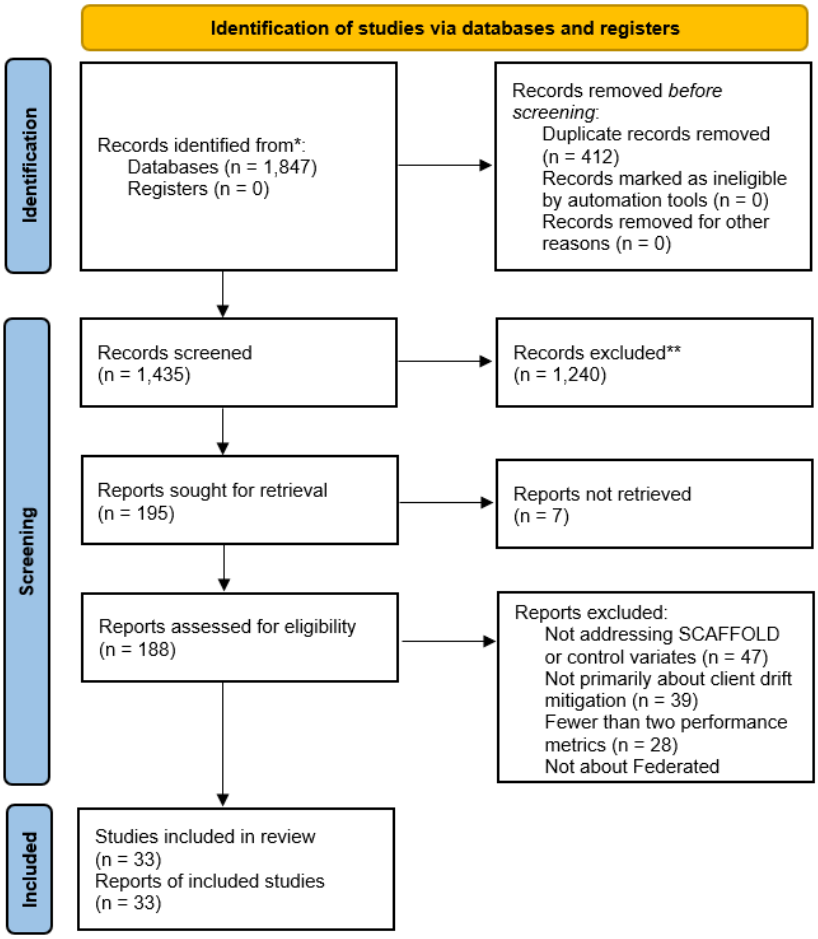


Figure 2.9: PRISMA 2020 Flow Diagram for Selection and Evaluation of Studies. (Source: Adapted from [12].)

Data were extracted from the 33 studies using a spreadsheet template recording the database of origin, paper code, title, authors, year, methods, results, conclusions, and stated future work. A thematic analysis identified recurring patterns, and the data were

synthesised through frequency counts and tabular summaries organised against the three research questions, forming the basis for the findings below.

2.7.2 Approaches Identified for Extending Scaffold (RQ1)

The first research question sought the approaches used to extend Scaffold for client-drift mitigation. Of the 33 included studies, 32 explicitly extended a Scaffold mechanism (one offered a theoretical analysis without an extension), yielding nine distinct approaches. Table 2.2 lists them with study counts and references; each is discussed below.

Table 2.3: Scaffold Extension Approaches for Client Drift Mitigation.

Approach	**No. of Studies**	**Percent (%)**	**References**
Variance reduction via gradient estimation techniques	11	34	[49], [50], [51], [52], [53], [54], [55], [56], [57], [58], [59]
Integration with advanced optimisation algorithms	8	25	[60], [61], [62], [63], [64], [65], [66], [67]
Partial client participation handling	3	9	[68], [69], [70]
Hybrid framework integration and domain adaptation	3	9	[71], [72], [73]
Privacy and security enhancements	2	6	[74], [75]
Communication efficiency and channel-aware methods	2	6	[76], [77]
Client similarity-based grouping	1	3	[78]
Enhanced bias and drift correction mechanisms	1	3	[79]
Theoretical analysis of existing control variates	1	3	[80]
Total	**32**	**100**	

Variance reduction via gradient estimation was the most common approach (eleven of the thirty-two studies), extending Scaffold's control variates with SVRG, SARAH, and SAGA estimators from the families examined in Section 2.4.3. An SVRG-style correction anchored on a frozen global optimiser state is used in [49], and within a

nested proximal loop for non-convex objectives in [54]. SARAH-style recursive reduction with random model selection, breaking standard Scaffold's communication barrier, appears in [52]. SAGA-style storage and reuse of each client's latest update removes partial-participation variance in [53] and, combined with SARAH recursion to avoid periodic full-gradient computation, reaches near-optimal complexity in [56], [58]. Further variants aggregate fresh and stale updates from all clients at no extra communication cost [50]; combine proximal historical-gradient mappings with inertial Douglas–Rachford splitting [51]; pair persistent per-client gradient statistics with episodic clipping to converge under subsampling, heterogeneity, and unbounded smoothness together [55]; track parameter–control-variate variances and covariances to prove simultaneous logarithmic communication and linear speed-up [57]; and apply a REINFORCE Leave-One-Out estimator that forms the baseline from the remaining gradients [59].

Integration with advanced optimisation algorithms was the second most common approach (eight studies). The correction term is replaced with a PID controller in [60], linking to the feedback-control paradigm in Section 2.5.7. Control variates are coupled with Sharpness-Aware Minimisation in [61] and Gradient Norm-Aware Minimisation in [63] to improve generalisation, and an Anderson Acceleration step approximating Newton–GMRES directions without the Hessian is added in [62]. Momentum lowers gradient variance and anchors local updates in [64], while client-side momentum with adaptive gradient normalisation gives a parameter-free algorithm for arbitrary heterogeneity in [66]. Federated ADMM is recast as a variance-reduction scheme with adaptive step sizes controlling the reduction level and global-model bias in [65], and Scaffold is paired with an accelerated primal-dual framework for distributionally robust optimisation in [67].

Partial client-participation handling was addressed by three studies, since Scaffold's full-participation assumption is unrealistic in practice. Client gradients and control variates are scaled by the participation rate to curb stale-variate effects in [68]; periodic or cyclic participation is handled through amplified updates and long-range control variates in [69]; and applying control variates only to the final classification layers improves communication efficiency and convergence while preserving feature diversity in [70]. Hybrid framework integration and domain adaptation also drew three

studies: model-contrastive learning with adaptive control variates for skew across Human Activity Recognition clients [71]; a unified scheme combining Scaffold's control variates with FedAvg averaging and a FedProx proximal term under accuracy-weighted aggregation on non-IID data [72]; and a transfer of the control-variate idea into federated fuzzy C-means clustering, with a co-initialisation module managing statistical and systems heterogeneity [73].

Privacy and security enhancements were addressed by two studies, as Scaffold transmits both model updates and control variates. Two time-varying noise-allocation schemes improve convergence while meeting total differential-privacy requirements in [74], and multi-key homomorphic encryption protects both parameters and control variates in transit, reweighting aggregation when users disconnect, in [75].

Communication-efficiency and channel-aware methods were the focus of two studies. MMSE estimation corrects simultaneously for data-driven client drift and wireless channel noise in [76], while a reformulated Scaffold transmitting a single, compressible increment variable with local momentum halves uplink communication in [77]. Client similarity-based grouping appeared in a single study, using the Pearson correlation coefficient to cluster clients with correlated updates before drift correction and so reduce local–global divergence [78]. This is the only similarity-based approach in the entire corpus, a finding whose significance is developed in Subsection 2.7.5. Enhanced bias and drift-correction mechanisms were addressed by one study, in which dual variables act as control variates accumulating historical gradient information to provide adaptive bias correction beyond the gradient-difference tracking of standard control variates [79].

Finally, one study offered a theoretical analysis rather than an extension, showing that Scaffold's global parameters and control variates jointly form a Markov chain converging geometrically to a unique stationary distribution in the Wasserstein distance, and that Scaffold still retains a higher-order bias from stochastic gradient noise, a result with implications for every extension built upon it [80].

2.7.3 Limitations of Existing Approaches (RQ2)

The second research question examined the limitations of these approaches. Organised by approach below, they map the practical and theoretical obstacles that motivated the design choices of CosScaffold (Chapter Three).

For variance reduction via gradient estimation, several limitations recur. Full-batch gradient evaluations at each global iteration become prohibitive on large local datasets [54]; large local minibatches may be infeasible for data-scarce clients [56]; and per-client auxiliary storage trades memory for communication efficiency [58]. Communication and computational overhead, with poor performance on sparse problems, is reported in [49]. The benefits are heterogeneity-limited: gains hold only under small heterogeneity and otherwise degrade to non-local rates [52], with communication advantages collapsing to centralised rates as Hessian heterogeneity approaches the smoothness constant [58]. Performance also fails under extremely sparse participation, where stale updates lose correlation with current gradients [50], and client-drift error is left unaddressed, persisting at O(1/T), in [53]. Other constraints include increasingly conservative learning rates as clients or rounds grow [55], a restriction to linear stochastic-approximation rather than general non-convex problems [57], the absence of explicit regularisation toward the global optimum [51], and anchoring on a single estimator (RLOO) with single-dataset, single-architecture evaluation [59].

For integration with advanced optimisation algorithms, the dominant limitation is hyperparameter sensitivity: PID-controller performance depends on precise gain tuning and can degrade results, with no selection guidance [60], and theoretical convergence requires tuning multiple interdependent, problem-specific hyperparameters [64]–[65]. Non-convexity is a further concern, with Anderson Acceleration trapped at critical points as gradient norms shrink [62] and strong-convexity assumptions limiting applicability to deep networks [67]. Overhead also recurs, doubled communication from transmitting parameters and control variates each round [61], and doubled computation from two gradient evaluations per step with no convergence guarantee [63]. Gradient normalisation, while enabling parameter-free operation, discards magnitude information and so depends critically on momentum [66].

For partial client-participation handling, the limitations reflect a tension between coverage and efficiency: transmitting both scaled and unscaled gradients adds overhead that contradicts the efficiency goal [68]; known, equal-expectation participation windows are required, precluding dynamic populations or clients freely joining and leaving [69]; and the layer split for variance reduction must be chosen manually per architecture, with no principled rule [70]. For hybrid framework integration and domain adaptation, the issues centre on distribution skew, scale, and complexity: severe degradation under label skew (up to a 10.75% F1-score drop), because both components rely on a global model that becomes biased when label sets differ across clients [71]; evaluation confined to simple benchmarks (MNIST, CIFAR-10) with only 100 clients, leaving scalability untested [72]; and added algorithmic complexity demanding careful tuning of local and global step sizes [73].

For privacy and security enhancements, the privacy–performance tension persists: one scheme balances but does not resolve the conflict, requiring early stopping and lower accuracy for its privacy guarantees [74], while another secures control variates without improving Scaffold itself, leaving the full-participation requirement and transmission overhead intact [75]. For communication-efficiency and channel-aware methods, restrictive assumptions and narrow validation dominate: a Gaussian assumption for local weights and gradients is suboptimal for real, non-Gaussian data [76], and experiments limited to small datasets (MNIST, Fashion-MNIST) with a two-hidden-layer network leave large-scale, modern-architecture behaviour unclear [77]. For client similarity-based grouping, the Pearson correlation coefficient cannot separate benign similarity (related data) from malicious similarity (a shared attack), creating a vulnerability that adversarial clients could exploit [78]. This directly motivated the choice of cosine similarity, rather than Pearson correlation, in CosScaffold.

For enhanced bias and drift correction, the adaptive mechanism works in moderate heterogeneity but needs complex tuning, imposes client-side overhead, and becomes unreliable under extreme non-IID distributions, concept drift, or adversarial conditions, where historical gradient accumulation breaks down [79]. For the theoretical analysis, Scaffold corrects most of the heterogeneity bias yet retains a higher-order bias from stochastic gradient noise that scales with the step size divided

by the number of clients and does not vanish beyond a threshold client count, leaving truly unbiased stochastic federated algorithms an open problem, a residual that applies to the base mechanism on which all other approaches build [80].

2.7.4 Scaffold Characteristics Extended (RQ3)

The third research question identified which characteristics of the Scaffold protocol are extended or modified for client-drift mitigation. Across the 30 studies that targeted a specific characteristic (excluding the theoretical analysis and two framework-level studies), twelve distinct characteristics emerged, including two combined categories. Table 2.3 reports these characteristics, the number of studies addressing each, and the corresponding references.

Table 2.4: Scaffold Characteristics Extended for Client Drift Mitigation.

Scaffold Characteristic	**No. of Studies**	**Percent (%)**	**References**
Variance Reduction	11	37	[78], [68], [60], [71], [51], [55], [69], [56], [57], [58], [59]
Drift Correction	3	10	[76], [61], [63]
Gradient Correction	3	10	[64], [66], [73]
Bias Correction	2	7	[79], [67]
Server Aggregation Strategy	2	7	[72], [49]
Variance Reduction and Drift Correction (combined)	2	7	[49], [70]
Variance Reduction and Bias Correction (combined)	2	7	[65], [52]
Gradient Estimation	1	3	[54]
Local Update Mechanism	1	3	[62]
Communication Mechanism	1	3	[77]
Privacy Budget allocation	1	3	[74]
Correlation Term	1	3	[50]
Total	**30**	**100**	

2.7.5 Key Findings and Research Gap

Three synthesis themes shaped CosScaffold (Cosine Similarity extended Scaffold). First, the field has concentrated on Scaffold's control-variate mechanism: variance reduction via gradient estimation (34% of the empirical sub-corpus, n = 32) and integration with advanced optimisation algorithms (25%) together account for 59% of

the work, while privacy and security (6%), communication efficiency (6%), and client similarity-based grouping (3%) remain under-investigated.

Second, several limitations cut across approaches. Communication and computational overhead was the most frequently reported [49], [54], [61], [63], [68], [75], concerning, since Scaffold already doubles FedAvg's communication cost. Hyperparameter sensitivity and tuning complexity limited practical applicability [60], [64]–[65], [73], [79], and restrictive theoretical assumptions (strong convexity, Gaussian or linear formulations) limited generalisability to deep networks [57], [62], [67], [76]. Performance also degraded under high heterogeneity [52], [58], [71], and validation at scale was limited [59], [72], [77]. These point to communication-efficient, adaptively tuned designs with relaxed assumptions and large-scale validation.

Third, and most directly motivating this study, similarity-based approaches were underexplored. Only one of the thirty-two extending studies used a similarity measure, the Pearson correlation coefficient, to group clients [78], and its own limitation showed that this measure cannot separate healthy similarity (well-trained models) from malicious similarity (compromised ones). This is a notable gap: unlike variance- or bias-correction methods that act after drift occurs, similarity measures can proactively detect and quantify client-to-global directional divergence before correction.

Adding a similarity measure as a third channel alongside the server and client control variates of Section 2.3.3, operationalised here as a per-client multiplicative modulator of the local learning rate rather than a formal control variate, could give a per-round signal of client-to-global alignment that modulates correction intensity. Aligned clients would receive minimal correction, preserving useful local information; divergent ones stronger correction, improving on Scaffold's uniform strategy (Subsection 2.3.3), cutting overhead, and improving robustness to heterogeneous data.

The review thus recommended similarity measures for this role. Four candidates were considered: cosine similarity, which captures directional alignment irrespective of magnitude; Euclidean distance, which quantifies the magnitude of gradient

divergence; Centred Kernel Alignment (CKA), a representation-level measure; and Jaccard similarity, adaptable to the sparsity patterns of large-model gradient updates. Cosine similarity was judged most promising, owing to its scale invariance, computational efficiency, and natural fit with the directional formulation of gradient drift established in Section 2.3.4. This evidence-based finding underpinned CosScaffold (Chapter Three), addressing Objectives Two and Three.

2.8 Existing Evaluation Techniques and Benchmarks

How the federated learning community evaluates and compares techniques matters as much as the algorithms themselves. Since FedAvg [1], evaluation has matured from accuracy on a single non-IID partition into multi-dimensional frameworks covering convergence, communication cost, statistical heterogeneity, client fairness, and robustness to partial participation. This section reviews the metrics most commonly reported and the standardised benchmarks used to assess client-drift mitigation techniques, establishing the methodological basis for the experimental design in Chapter Three.

2.8.1 Evaluation Metrics

FL evaluation metrics fall into four categories: core performance, drift-specific, heterogeneity quantification, and fairness. The chosen subset shapes what can be concluded about an algorithm.

Core performance metrics centre on global test accuracy on a held-out set. The convention of plotting accuracy against communication rounds, with linear interpolation to the rounds needed for a target accuracy, was established in [1]; this rounds-to-target measure is now the primary efficiency indicator, with the original FedAvg work reporting convergence in an order of magnitude fewer rounds than synchronised SGD [1]. Communication cost is measured coarsely (synchronisation rounds) and finely (bytes transmitted): Scaffold [2] needs roughly twice the per-round communication of FedAvg because it also sends control variates, whereas FedDyn [5] matches FedAvg exactly. Communication was formalised as the principal constraint in [81], whose structured and sketched updates reduce uplink cost. Papers increasingly add local compute cost, wall-clock convergence time, and loss curves; precision,

recall, F1-score, and area under the ROC curve supplement accuracy under class imbalance, as in healthcare and fraud detection [20].

Drift-specific metrics quantify the divergence between local and global parameters central to heterogeneous FL. The most direct is weight divergence, the ℓ_2 norm $\|w_{local} - w_{global}\|$. In [3] it was shown to grow with heterogeneity and correlate strongly with accuracy loss (up to a 55% drop on CIFAR-10 under the most skewed partition), and to be upper-bounded by a function of the Earth Mover's Distance between client and population label distributions, the learning rate, and the gradient norms. The principal theoretical measure is the bounded gradient dissimilarity assumption of [2] (Section 2.3.2): the (G, B)-BGD assumption posits non-negative constants G and B such that the average squared gradient norm across clients is bounded by $G^2 + B^2\|\nabla f(w)\|^2$ for all w, with G capturing absolute and B relative heterogeneity. A related B-local dissimilarity was used for FedProx [4], and a simpler bound on the expected squared client–global gradient difference by ζ^2 is widely adopted [22].

A refinement in [82] showed that bounded gradient dissimilarity can be arbitrarily large even when FedAvg performs well. It introduced the average drift at optimum, $\rho^2 = \|E_c[\nabla F_c(w^*)]\|^2$ where w^* is the global optimiser, proving ρ^2 is never greater than ζ^2 and is approximately zero on practical datasets such as FEMNIST and StackOverflow even when ζ^2 is large [82],helping explain FedAvg's empirical success. A further measure central to this study is cosine similarity between client gradient updates, used in [83] to bipartition clients with conflicting directions and later extended to client selection [7], Byzantine-robust aggregation [8], and gradient-based defences [9] (Section 2.5.3).

Heterogeneity quantification metrics address how non-IID-ness is built and measured. The field has moved from the pathological construction of [1], where the dataset is sorted by label and each client gets only k classes, to Dirichlet-based partitioning, in which each client's label proportions are drawn from a Dirichlet distribution with concentration parameter α. Established as standard in [84], performance degrades smoothly as α falls from roughly 0.01 (about one class per client) to the IID limit;

common values span {0.01, 0.05, 0.1, 0.3, 0.5, 1.0}, with studies typically sweeping two or three. The most systematic treatment, NIID-Bench [32], implemented six partition strategies: label distribution skew with a fixed class count, Dirichlet-based label skew, quantity skew, feature distribution skew, noise-based feature skew, and an IID baseline. Divergence is quantified by the Earth Mover's (Wasserstein-1) Distance, shown in [3] to correlate strongly negatively with performance, plus Kullback-Leibler and Jensen-Shannon divergence, Maximum Mean Discrepancy, and the Kolmogorov-Smirnov statistic [21].

Fairness metrics expose the per-client disparities that aggregate accuracy hides. The simplest is accuracy variance or standard deviation across clients, lower being more equitable. The q-Fair Federated Learning (q-FFL) objective [85] uses an α-fairness-inspired optimisation with a non-negative scalar q, where higher q emphasises higher-loss clients; q = 0 recovers FedAvg and large q approaches minimax fairness, reporting accuracy variance, bottom-10% device accuracy, and the best-to-worst accuracy ratio [85]. More extreme, the Agnostic Federated Learning of [86] solves the minimax problem min_w max_i F_i(w) to protect the worst-served client. Jain's Fairness Index [87] is adapted as JFI = $(\sum x_i)^2 / (n \cdot \sum x_i^2)$, taking values in [1/n, 1] and maximised when all clients are served equally. The Ditto framework [35] revealed a robustness–fairness tension, using personalised multi-task learning with a proximal regulariser to cut per-client variance while raising absolute accuracy. Together these four families,performance, drift, heterogeneity, and fairness,define the multidimensional profile against which a drift-mitigation technique must be assessed.

2.8.2 Benchmarks

Alongside these metrics, the community has built benchmark suites and standardised datasets for fair comparison. The earliest and most influential is LEAF [88] (2018), a modular framework of open-source federated datasets, a unified protocol, and reference implementations. Its purpose-built datasets,FEMNIST (EMNIST characters partitioned across writers), Shakespeare (next-character prediction by play role), Sent140 (tweet sentiment by user), and CelebA (face attributes by celebrity),plus synthetic tasks, capture heterogeneity from natural data-generating processes rather than artificial partitioning, and are now the de facto reference set. The rounds-to-target metric of [1] is operationalised across them by LEAF [1], [88].

Later suites tackled two LEAF limitations: weak support for systems heterogeneity (client compute and network variation) and poor scaling beyond a few hundred clients. FedScale [89] (2022) broadened coverage to image classification, object detection, language modelling, and speech recognition, each with a unified protocol on real-world splits and per-client reporting, and added a scalable runtime emulating realistic device heterogeneity from production traces [89]. NIID-Bench [32], companion to the partition-strategy taxonomy above, focuses on how different heterogeneity constructions affect performance, supporting the six strategies listed earlier; it has been used to compare the robustness of FedAvg, FedProx, Scaffold, MOON, and related techniques under controlled variation of the heterogeneity parameter [32].

Several datasets recur often enough to be de facto standards. CIFAR-10 and CIFAR-100, though originally centralised, are commonly partitioned across simulated clients by Dirichlet allocation with α typically in $\{0.1, 0.3, 0.5\}$. EMNIST and FEMNIST, partitioned by writer, remain the canonical handwritten-character benchmarks and featured in the original Scaffold evaluation [2]. Shakespeare gives a sequence-modelling benchmark with natural per-role heterogeneity, while FedScale [89] added StackOverflow next-word prediction, OpenImages, and Speech Commands for larger-scale work. Most recent Scaffold extensions (Section 2.6.3) use at least two of these, though varying partition strategies and concentration parameters hamper direct comparison.

A persistent challenge is the absence of a single universally adopted platform. LEAF [88], FedScale [89], NIID-Bench [32], FedML, FedEval, and others trade off scale, coverage, systems realism, and ease of use differently, and none holds the role MLPerf plays in centralised machine learning. Studies of Scaffold and its extensions therefore report across two or three configurations, most often using rounds-to-target accuracy, communication cost, and,less consistently,per-client fairness. This thesis grounds its evaluation in the original Scaffold protocol [2] for direct comparability with the published baseline. CosScaffold (Cosine Similarity extended Scaffold) is evaluated across three configurations: a convex testbed (logistic regression on the EMNIST Balanced split), a small non-convex testbed (two-layer multilayer perceptron on the same split), and a non-convex image-classification testbed (small convolutional neural

network on CIFAR-10). A pre-registered synthetic quadratic configuration for early convergence validation lies outside the principal Configuration A/C/D evaluation in Chapter Four. Heterogeneity uses the label-sorted-with-IID-mixing scheme of [2], with 100 clients, sampling fraction 0.2, and five local epochs per round; Dirichlet partitioning [84] is left as future work. Four metric families are tracked per run,convergence (rounds-to-target accuracy), communication cost (bytes), drift diagnostics (control variate norms and weight divergence), and convergence stability,consistent with the standards above.

2.8.3 Validation Techniques in Federated Learning

Evaluating FL algorithms demands validation beyond that used for centralised machine learning, because their distributed, stateful, non-IID character introduces variability, bias, and confounding that single-machine benchmarking does not address. This subsection reviews the principal techniques underpinning the validation framework of Chapter Three.

Cross-Validation and Data Partition Strategies. Classical k-fold cross-validation cannot be applied directly, because the client partition reflects the natural distribution of users or institutions and cannot be rearranged per fold. Validation instead uses a held-out global test set, evaluated by the server each round [1], [15]. This convention, established in [1] and adopted across LEAF [88] and NIID-Bench [32], was extended in [32] to a federation-level train/validation/test split. This study adopts that partition: validation for per-algorithm grid-search tuning, and a separate test split for all comparisons in Chapter Four.

Ablation Studies. Ablation isolates a design choice by removing or replacing it and comparing against the full system. In FL it has validated components such as the control variate in Scaffold [2], the proximal regulariser in FedProx [4], the contrastive loss in MOON [30], and the drift variable update in FedDyn [5]. This study uses a pre-registered five-arm ablation (Section 3.4.2) comparing full CosScaffold against identity (recovering vanilla Scaffold), random, sign-only, and anti-modulators. This is more rigorous than the usual with-versus-without test, separately checking whether the magnitude and direction of the cosine signal are both necessary, ruling out that any per-client step-size variation or signed modulation alone would suffice.

Statistical Testing and Multiple-Comparison Control. FL has historically reported with limited rigour: mean or median accuracy over a few seeds, with no significance testing or confidence intervals, so genuine gains are hard to separate from initialisation or partition noise. This was flagged in [32], since single-seed results are not reliably reproducible; recent work reports multiple seeds and bootstrap intervals [55]. This study pre-registers all tests, applies Holm-Bonferroni correction [90] to control the family-wise error rate at $\alpha = 0.05$ per hypothesis family, selects between a paired t-test and a paired Wilcoxon signed-rank test [94] via an automated Shapiro-Wilk check [95], and builds confidence intervals by bootstrap [93] with 10,000 resamples.

Pre-registration and Reproducibility. Pre-registration,publicly committing to a hypothesis, design, and analysis plan before collecting data,is standard in clinical research and psychology but rare in machine learning; it prevents post-hoc selection of favourable hyperparameters, datasets, metrics, or comparators from inflating results. Reproducibility is itself a pervasive problem: audits of major AI venues find fewer than one in five papers give enough detail for independent replication. This study anchors pre-registration cryptographically: every design decision,comparator set, modulator hyperparameters, ablation arms, hypothesis families, statistical procedures, stable-window thresholds, and seed list,was frozen in a JSON document whose SHA-256 digest was recorded before any data were collected. Being purely cryptographic and needing no external authority, this is stronger than conventional platforms.

Expert Review and Survey-Based Validation. Expert review adds a qualitative dimension, soliciting structured specialist judgments on a technique's soundness, novelty, and practical relevance. In FL it has assessed real-world deployability in healthcare consortia [20], mobile computing [1], and financial services [14]. It is noted in [20] that expert consensus matters especially in regulated domains such as medical imaging, where only specialists can judge the gap between benchmark and clinical utility. This study pairs expert analysis with the systematic review: the SLR catalogue of extension approaches, characteristics, and limitations was reviewed against expert analysis to validate the evidence-based choice of cosine similarity for CosScaffold, forming the mixed-methods qualitative dimension discussed in Section 3.2.

Construct Validity and Measure Discrimination. Construct validity asks whether metrics measure what they claim. A well-known FL concern is gradient dissimilarity: the (G, B)-BGD assumption of [2] and the ζ^2 bound both measure the spread of client gradients, yet [82] showed these can be arbitrarily large even when FedAvg converges well, questioning their validity as performance predictors. This motivates complementary measures. This study computes both cosine similarity and Euclidean distance between client probe directions and the previous aggregated update for every (round, client) pair, testing their correlation across 1,960 paired observations. That they explain less than 1% of each other's variance (Pearson $r = 0.054$, Spearman $\rho = 0.098$) confirms that cosine similarity captures a directional alignment construct distinct from the magnitude-based Euclidean one.

Convergence-Stability Validation. Fixed communication-round budgets pose a challenge: a run not yet converged at the budget can look identical to one that has stalled or diverged, misleading any comparison based on terminal accuracy alone [32]. Recent papers report full learning curves rather than endpoints [2], [4], [55], and NIID-Bench [32] proposed tracking per-round accuracy variance as a stability indicator, since high late-training variance can signal oscillation. This study introduces a pre-registered stable-window classifier (Subsection 3.7.4) computing the standard deviation and linear slope of the test-accuracy trajectory over each run's final 20 rounds, classifying it as converged, still-learning, oscillating, or diverging. Adding a directional slope to the variance indicator distinguishes steady improvement from stagnation and diagnoses whether differences are genuine or artefacts of budget truncation.

2.9 Chapter Summary

This chapter has established the theoretical, methodological, and empirical foundations on which the present study rests. Federated learning [1], [15] was introduced as the broader paradigm, client drift [2], [3] was identified as its central optimisation pathology under heterogeneous client data, and the formal apparatus of FedAvg [1] and Scaffold [2] was developed to characterise it. A survey of the existing drift-mitigation techniques and the design paradigms on which they rest set the stage for the PRISMA 2020-compliant systematic literature review reported in Section 2.6,

which catalogued nine extension approaches, the recurring limitations across the 33 included studies, and the twelve Scaffold characteristics most commonly targeted for modification. The review revealed that similarity-based mechanisms remain markedly underexplored in the Scaffold literature: only one of the thirty-two relevant studies employed any similarity measure, and that single study used Pearson correlation rather than cosine similarity and acknowledged its own inability to distinguish healthy from malicious similarity. Cosine similarity, with its scale invariance, low computational cost, and natural fit with the directional formulation of gradient drift, offers a principled basis for an adaptive modulator of the local learning rate. This evidence-based gap directly motivates the design of the CosScaffold (Cosine Similarity extended Scaffold) technique presented in Chapter Three.

Table 2.5: Chapter Summary

Approach / Baseline	**Client Drift Mitigation Mechanism**	**Reported Performance**	**Relation to CosScaffold / Gap Addressed**
FedAvg [1]	None -assumes approximately IID data	Baseline; accuracy drops up to 55% under extreme non-IID skew [3]	Establishes the baseline drift problem CosScaffold, via Scaffold, is designed to correct.
FedProx [4]	Penalises parameter deviation from global model (magnitude-based, undirected)	~22% average accuracy gain over FedAvg under heterogeneity; convergence still degrades with G	Motivates the need for a *directional* (not isotropic) correction -the core design choice in CosScaffold.
Scaffold [2]	Corrects gradient-level discrepancy directly; convergence independent of dissimilarity (G, B)	Matches/beats centralised SGD complexity; unstable under some extreme non-IID configurations [32]	Direct base algorithm: CosScaffold retains its correction and guarantee, adds cosine-similarity-

			based local-rate modulation.
FedDyn [5]	Server-side dynamic penalty aligning local/global optimality conditions	Convergence independent of heterogeneity; unbounded dual-variable growth causes instability [10]	Cautionary precedent for unbounded correction terms; CosScaffold's modulator is explicitly clipped/bounded to avoid this failure mode.
FedDC [6]	Parameter-level correction of cumulative local–global gap	Outperforms Scaffold on CIFAR-10/100 empirically	Illustrates the trade-off of correcting at the parameter level rather than the gradient level; CosScaffold preserves Scaffold's gradient-level, dissimilarity-independent guarantee instead.
FedNova [23]	Corrects *aggregation-objective mismatch* from unequal local step counts, not gradient-direction drift	Subsumes FedAvg/FedProx as special cases; removes a distinct systems-heterogeneity bias	Complementary, orthogonal concern (systems heterogeneity) to the statistical-heterogeneity drift CosScaffold targets.
BN-Scaffold [11]	Extends control-variate correction to batch-norm statistics drift	Shows standard Scaffold's correction is insufficient for BN-induced drift	Reinforces that Scaffold's *static, uniform* correction has multiple identified blind spots, supporting

			the case for an adaptive extension such as CosScaffold.
Personalisation methods (Per-FedAvg [33], pFedMe [34], Ditto [35])	Accepts heterogeneity rather than correcting it; optimises per-client models	Strong per-client accuracy; does not improve the single shared global model	Different objective (personalisation vs. shared-model convergence) - reviewed for context but excluded as an experimental baseline, since it does not target the same drift-correction problem as CosScaffold.
Distillation methods (FedDF [36], FedNTD [37])	Constrains output/decision-boundary behaviour rather than gradients or parameters	Supports heterogeneous client architectures; resists drift on non-true-class logits (FedNTD)	Reviewed to contextualise the drift problem; not included as an experimental baseline since CosScaffold's contribution is specifically at the gradient/local-step level.

CHAPTER THREE
METHODOLOGY

3.1 Introduction

This chapter sets out the methodology used to design, implement, and evaluate CosScaffold (Cosine Similarity extended Scaffold), an enhanced Scaffold technique for mitigating client drift in Federated Learning under non-IID data. The extension introduced earlier is realised as a cosine-similarity control mechanism that scales each client's local learning rate by how closely its update direction agrees with the previous round's aggregated server update. This further curbs the client drift the original Scaffold algorithm of [2] was built to address, while reducing exactly to vanilla Scaffold as a no-information baseline. Because the study extends an established technique rather than proposing a new theory, every methodological choice follows one principle: adopt the experimental protocol of [2] wherever it is sound and directly comparable, make explicit what that paper left implicit, and add what a rigorous master's-level empirical contribution now demands.

This principle takes shape as a three-pillar discipline applied throughout the chapter. The first pillar, grounding, anchors the work in the original Scaffold study so that results remain directly comparable: dataset choices, client population, partition scheme, comparator set, and metric definitions are inherited wherever possible without compromising rigour. The second pillar, precise specification, fixes the decisions the original paper left open to challenge,the exact formulation of the modulator, a convergence definition based on a stable-window classifier, the family-wise statistical procedures, and the cryptographically anchored pre-registration of hypotheses and analyses. The third pillar, principled extension, adds what the original omitted but is now expected: a five-arm ablation isolating the modulator's contribution, a construct-validity check comparing cosine and Euclidean alignment, and a failure-mode classifier explaining why models converge or fail. The pillars are sequential and obligatory; omitting any one would leave a weakness an examiner could legitimately exploit.

The chapter then proceeds in sequence. It first establishes the research design and the phased process the study followed, then specifies the design of the enhanced Scaffold technique,the cosine-similarity control mechanism and the formulation of adaptive

learning rates and control variates,thereby addressing the second research objective. The evaluation framework follows, covering the simulation environment, the benchmark datasets, the modelling of data heterogeneity, and the implementation choices that turn it into running code. The descriptive and inferential analysis plan is then set out, including the four metric families and the pre-registered hypothesis tests, addressing the third objective: evaluating the technique against its baselines. The chapter closes by examining threats to reliability and validity and their safeguards, considering the ethical aspects of the study, and summarising the discussion.

3.2 Research Design

This study adopts an experimental, quantitative research design that integrates a quantitative experimental strand with qualitative strands, namely a PRISMA 2020-compliant systematic literature review (SLR) and expert review of the SLR's recommendations, in order to address the three research objectives. The quantitative strand employs controlled simulation as its overall research strategy. The phenomena under investigation, namely gradient dynamics, client drift, convergence behaviour, and communication efficiency in federated optimisation, exist as objective and mathematically governed realities that are independent of the observer, and they are directly measurable through the metrics reported by the experimental runs. The study is therefore quantitative in nature and proceeds through deductive hypothesis testing within a pre-registered analytical framework. Hypotheses are derived from the theoretical analysis of client drift presented in Chapter Two (Sections 2.2.4 to 2.2.5) and from the empirical regularities established by [2] and [3] for the federated non-IID setting. The CosScaffold (Cosine Similarity extended Scaffold) technique developed in Chapter Two is treated as the experimental intervention; FedAvg [1], FedProx [4], Scaffold [2], FedDyn [5], and ScaffoldComp [77], together with centralised SGD as a non-federated reference, constitute the controlled comparator conditions.

The research is applied and developmental in type. It applies an established theoretical mechanism, namely variance reduction via control variates as in Scaffold [2], and develops a principled refinement of it through cosine modulation of the local learning rate. The contribution is empirical and methodological rather than purely theoretical: the study investigates whether a specific, well-motivated modification produces a

measurable improvement over the published Scaffold baseline, and under which conditions such improvement is observed. The unit of analysis is the federated learning run, defined as a single execution of one algorithm on one dataset partition with one random seed for a fixed number of communication rounds. Each run produces a trajectory of communication-round-indexed metrics, namely training loss, test accuracy, communication cost, drift diagnostics, and cosine alignment statistics, together with a terminal summary that includes a stable-window classification of the run's convergence behaviour (Section 3.7.4). Each unique combination of algorithm, dataset configuration, similarity level, and seed therefore generates one observation in the analysis.

The study takes the form of a specified-and-observed experiment. The algorithm, the experimental conditions, and the pre-registered analytical procedure are fixed in advance, but the data are not generated by hand: they are produced automatically by the federated simulation, and the cryptographic pre-registration commitments detailed in Section 3.8 prevent post-hoc modification of the analytical plan. This separation between specification and observation, enforced by the digest-anchored protocol, is the principal mechanism by which the design controls for analytical degrees-of-freedom and confirmation bias in the interpretation of results.

3.3 Research Process and Phases

The study was conducted in six sequential phases, each addressing a defined set of methodological deliverables and each providing the inputs that the subsequent phase required. The phases follow the standard arc of an empirical machine-learning study (problem formulation, literature synthesis, design, implementation, evaluation, and analysis), but they are tied here explicitly to the three Research Objectives so that the research process can be audited against the objectives it was meant to satisfy. Table 3.1 maps each phase to the objective it served and the principal deliverable it produced, and the prose that follows describes each phase in turn.

Table 3.1: Phases of the Research Process and their Mapping to the Research Objectives.

Phase	Activity / Description	Objective Served and Principal Deliverable
Phase 1	Problem formulation and scoping	Frames Objectives 1-3. Deliverable: problem statement, objectives, questions, scope (Chapter 1).
Phase 2	Systematic literature review (PRISMA 2020)	Objective 1. Deliverable: catalogue of 9 extension approaches, 12 Scaffold characteristics, evidence-based gap (Section 2.6).
Phase 3	Algorithm design of CosScaffold	Objective 2. Deliverable: cosine modulator, identity-clamping discipline, five-arm ablation framework (Section 3.4).
Phase 4	Implementation and cryptographic pre-registration	Supports Objective 3. Deliverable: simulation codebase, tuned hyperparameters, SHA-256-anchored analytical plan (Section 3.8).
Phase 5	Controlled-scale evaluation	Objective 3. Deliverable: per-round metrics across 7 algorithms x 3 configurations x 3 similarity levels x 2 seeds (Chapter 4).
Phase 6	Analysis, synthesis, and reporting	Completes Objective 3. Deliverable: convergence, accuracy, stability, and construct-validity findings; this thesis (Chapters 4-5).

Phase 1, problem formulation and scoping, established the research problem and the framework within which it would be addressed. The investigator examined the client-drift phenomenon as documented in the federated-learning literature [1], [2], [3], identified its consequences for convergence and communication cost in the non-IID regime, and formulated the three Research Objectives that organise the rest of the study. This phase produced the problem statement, the objectives and questions, and the scope and limitations recorded in Chapter One. It did not generate new empirical

data; its function was to delineate what the study would investigate, what it would not investigate, and against what intellectual context the contribution would be judged.

Phase 2, the systematic literature review, satisfied Objective One. The investigator conducted a PRISMA 2020-compliant review across seven electronic databases, identifying 1,847 records, screening them in three stages against pre-specified inclusion and exclusion criteria, and arriving at a final corpus of 33 studies. The corpus was synthesised against three review questions corresponding to the three sub-objectives of Objective One, producing a catalogue of nine extension approaches that prior researchers have applied to the Scaffold framework, the recurring limitations across the included studies, and the twelve Scaffold characteristics most commonly targeted for modification. The review exposed the specific evidence-based gap that motivates the work of subsequent phases, namely that similarity-based mechanisms are markedly underexplored in the Scaffold literature and that cosine similarity in particular has not previously been applied as an adaptive control variate in this setting. The full protocol, including search strings, screening logs, and quality-assessment outcomes, is reported in Section 2.6.

Phase 3, algorithm design, satisfied Objective Two. Starting from the gap identified in Phase 2, the investigator designed the CosScaffold (Cosine Similarity extended Scaffold) technique as a strict generalisation of the Scaffold algorithm of [2]. The design proceeded in three steps: the choice of signal, namely cosine similarity between a client's probe direction and the previous round's aggregated server update; the formulation of the modulator, namely a clipped affine map from cosine value to a per-client scaling factor for the local learning rate; and the identity-clamping discipline that ensures CosScaffold reduces exactly to vanilla Scaffold under no-information conditions, thereby preserving the convergence guarantees of the underlying algorithm at the boundary. The five-arm ablation framework that isolates the contribution of the modulator was also specified in this phase. The full design is presented in Section 3.4.

Phase 4, implementation and pre-registration, prepared the apparatus required for the empirical work and laid the methodological foundations for Objective Three. The investigator implemented CosScaffold and the six comparator algorithms in a single

federated-simulation codebase, verified the implementation against the Scaffold invariant in low-level tests, and tuned the local-learning-rate hyperparameters per algorithm-by-configuration cell. In parallel, the complete analytical plan, including the comparator set, modulator hyperparameters, ablation arms, four pre-registered hypothesis families, the stable-window convergence classifier, the statistical procedures, and the random seed list, was frozen in a single JSON document whose SHA-256 digest was recorded before any experimental data were collected. This pre-registration discipline, described in detail in Section 3.8, is the principal safeguard against the analytical degrees-of-freedom that would otherwise threaten the reliability of the comparative findings.

Phase 5, the controlled-scale evaluation, generated the empirical evidence on which Objective Three rests. The seven algorithms were executed across the three experimental configurations (Configuration A: logistic regression on EMNIST Balanced; Configuration C: a two-layer multilayer perceptron on EMNIST Balanced; Configuration D: a small convolutional network on CIFAR-10) and the three similarity levels ($s = 0$, $s = 10$, $s = 100$) for a 50-round communication budget under two random seeds per cell. Per-round metrics covering convergence, communication cost, drift diagnostics, and stability were logged to a manifest of experimental artefacts. The controlled-scale design is a deliberate methodological choice, explained in Chapter Four and discussed at greater length in Sections 1.7 and 3.5: it isolates algorithmic differences from the confounding variability that would arise in a longer, less standardised run, and produces directional evidence whose statistical confirmation under the full pre-registered protocol is identified as the primary future-work item in Section 5.4.

Phase 6, analysis and reporting, completed Objective Three and produced the thesis itself. The pre-registered analyses were executed on the artefacts generated in Phase 5: descriptive statistics for the convergence and accuracy metrics, the stable-window classifier for the convergence-stability analysis, and the cosine-versus-Euclidean correlation for the F4 construct-validity analysis. The findings were synthesised in Chapter Four, situated in the broader landscape of federated optimisation in Chapter Five, and the recommendations for practice, policy, and further research were derived from them. Throughout this phase, the cryptographic pre-registration digest

established in Phase 4 served as the audit trail against which every reported analysis can be verified by an examiner or independent researcher.

3.4 Design of the Enhanced Scaffold Technique (Objective 2)

3.4.1 Cosine Similarity-based Control Mechanism

The Scaffold algorithm of [2], formally examined in Subsection 2.3.3 of the literature review, addresses client drift by maintaining a server-level control variate c and a per-client control variate c_i for each client i, and by adjusting each client's local gradient by the difference (c − c_i) at every local step. This correction reduces the bias introduced by heterogeneous client objectives, but it does not adjust the magnitude of the local step on the basis of how well an individual client's update agrees with the global trajectory at the current point in training. In the heterogeneous setting, even after the Scaffold correction is applied, the directions of individual client updates can still vary substantially around the aggregated server update from one round to the next. Clients whose updates align with the most recent aggregated direction are providing information that is broadly consistent with the global trajectory and that the aggregation step will reinforce; clients whose updates oppose the aggregated direction are providing information that the aggregation step will partially cancel. Conventional Scaffold treats both kinds of clients identically by applying the same local step size η_l to all participating clients in all rounds.

The CosScaffold (Cosine Similarity extended Scaffold) enhancement introduces a per-client multiplicative modulator m_i that scales the local learning rate as η_eff,i = m_i · η_l. The modulator is derived from the cosine similarity between a probe of the client's local gradient direction and the aggregated server update from the previous round. The choice of cosine similarity rather than Euclidean distance is principled and is justified at length in Subsection 2.3.4 of the literature review: cosine similarity is scale-invariant, it operates exclusively on directional information, and it is bounded in the interval [−1, +1], properties that together make it a natural and computationally inexpensive instrument for the bounded, directionally-aware modulation that the Scaffold update rule requires. Under this mechanism, clients whose probe direction agrees with the aggregated server update receive an effective local step that is amplified, thereby reinforcing contributions consistent with the global trajectory; clients whose probe opposes the aggregated direction receive a dampened step,

thereby attenuating contributions that the aggregation step would otherwise partially cancel. The mechanism therefore enables each client's effective step size to respond continuously to the instantaneous severity of drift experienced at that point in training, rather than applying a uniform step size irrespective of alignment.

A critical design constraint of the modulation mechanism, fixed in the pre-registration, is that the modulator must reduce to the identity ($m_i = 1$) whenever no directional information is available. This identity-clamping requirement is engaged in two circumstances. The first is the initial communication round, in which the previous-round aggregated server update is undefined. The second is whenever a client's gradient probe or the previous aggregated update has a norm below a small numerical floor, in which case the cosine similarity is numerically undefined and any computed value would be dominated by floating-point noise. Under both conditions, the modulator falls back to the identity and the local update reduces exactly to the standard Scaffold update of Equation 11 in Chapter Two. CosScaffold (Cosine Similarity extended Scaffold) is therefore a strict generalisation of vanilla Scaffold: at the boundary conditions where no directional signal is available, it recovers Scaffold exactly, and the convergence guarantees of the original technique are preserved at those boundaries. This property is operationally essential for the ablation framework presented in Subsection 3.4.2: the first ablation arm sets $m_i = 1$ unconditionally and, by construction, recovers Scaffold.

3.4.2 Formulation of Adaptive Learning Rates and Variates

This subsection states the precise mathematical formulation of CosScaffold (Cosine Similarity extended Scaffold). The exposition proceeds in three parts: the Scaffold update rules inherited from [2]; the cosine similarity modulator that constitutes the extension; and the five-arm ablation framework that isolates the modulator's contribution. Throughout this subsection, N denotes the total number of clients, $S^l \subseteq \{1, \ldots, N\}$ denotes the set of clients sampled in communication round r, K denotes the number of local steps per round, η_l denotes the nominal local step size, and η_g denotes the global (server) step size.

Each sampled client $i \in S^l$ initialises its local model $y_i \leftarrow x$ from the current server model and performs K local SGD steps of the form:

$$y_i \leftarrow y_i - \eta_{eff,i} \cdot (g_i(y_i) + c - c_i) \quad \text{eq.19}$$

where $g_i(y_i)$ is a stochastic gradient evaluated on client i's local data, c is the server-level control variate, and c_i is client i's local control variate. Under conventional Scaffold the effective local step size $\eta_{eff,i}$ equals the nominal local step size η_l for every client and every round; under CosScaffold (Cosine Similarity extended Scaffold), $\eta_{eff,i}$ is replaced by the modulated value $m_i \cdot \eta_l$, where m_i is defined below. After completing the K local steps, each client computes its updated local control variate using Option II [2], which re-uses the gradients already computed during local training rather than incurring an additional gradient evaluation at the server model:

$$c_i^{+} \leftarrow c_i - c + (1 / (K \cdot \eta_{eff,i})) \cdot (x - y_i) \quad \text{eq.20}$$

The server then aggregates the client updates using the global step size η_g and updates the server-level control variate using the asymmetric (1/N) normalisation that preserves the Scaffold invariant $c = (1/N) \Sigma_i c_i$ under partial participation:

$$x \leftarrow x + (\eta_g / |S^l|) \cdot \Sigma_{\{i \in S^l\}} (y_i - x) \quad \text{eq.21}$$

$$c \leftarrow c + (1 / N) \cdot \Sigma_{\{i \in S^l\}} (c_i^{+} - c_i) \quad \text{eq.22}$$

Equations 19 to 22 are reproduced exactly from Algorithm 1 of [2] with Option II for the local control variate update. Under the design choice $m_i = 1$ for all i and all r, the CosScaffold (Cosine Similarity extended Scaffold) algorithm recovers Scaffold exactly. The Scaffold invariant $c = (1/N) \Sigma_i c_i$ was verified empirically during implementation: across all controlled-scale evaluation runs, the maximum deviation from this invariant was 1.86×10^{-9}, which is consistent with floating-point round-off and confirms that the Scaffold-family implementations used in this study are correct.

The cosine similarity modulator is constructed as follows. Before its K local steps in round r, each sampled client i computes a probe direction p_i by evaluating a single stochastic gradient at the server model and negating it: $p_i = -g_i(x)$. The probe is then compared against the previous round's aggregated server update $x_{aggr,prev}$ via the cosine similarity:

$$\cos_i = \langle p_i, x_{aggr,prev} \rangle / (\|p_i\| \cdot \|x_{aggr,prev}\|) \quad \text{eq.23}$$

where $\langle \cdot, \cdot \rangle$ denotes the Euclidean inner product and $\| \cdot \|$ denotes the L2 (Euclidean) norm of a vector. The modulator m_i is then constructed as a clipped affine function of the cosine value:

$$m_i = \text{clip}(1 + \alpha \cdot \cos_i, \eta_{min}, \eta_{max}) \quad \text{eq.24}$$

The hyperparameter values fixed at pre-registration are $\alpha = 0.5$, $\eta_{min} = 0.5$, and $\eta_{max} = 1.5$. These choices imply that the modulator can at most multiply the effective local step size by 1.5 (when cos_i = 1, the unclipped value would be exactly 1.5) or by 0.5 (when cos_i = −1). The identity value m_i = 1 corresponds to cos_i = 0, that is, to a probe direction orthogonal to the previous aggregated update. The choice of α = 0.5 is deliberately conservative: it confines the modulation envelope to a symmetric ±50% range around the nominal local step size, ensuring that the modulator cannot drive the effective step size into a regime substantially different from that on which Scaffold's convergence guarantees were derived.

The identity-clamping requirement specified in Subsection 3.4.1 is enforced as follows. In the initial communication round, the previous-round aggregated update x_aggr,prev is undefined; the algorithm sets m_i = 1 for all sampled clients in this round. In any subsequent round, if either $\|p_i\|$ or $\|x_{aggr,prev}\|$ falls below a numerical floor of 10^{-12}, the cosine similarity is treated as zero and m_i is set to 1. Both conditions cause CosScaffold (Cosine Similarity extended Scaffold) to fall back exactly to Scaffold for that client in that round, ensuring that the algorithm is well-defined under all numerical conditions and that no spurious modulation arises from floating-point noise on near-zero vectors.

To isolate the contribution of the cosine similarity modulator and to rule out alternative explanations for any observed performance change, the algorithm is instantiated under five ablation arms, each fixed in the pre-registration before any experimental data was generated. Table specifies the five arms and the purpose served by each in the inferential framework.

Table 3.2: Five-Arm Ablation Framework for Isolating the Cosine Modulator's Contribution.

Arm	**Definition of m_i**	**Purpose in the Inferential Framework**
Arm 1	m_i = 1 (identity, unconditional)	Recovers vanilla Scaffold exactly. Serves as the no-information baseline against which the gain attributable to any directional modulator is measured.
Arm 2	m_i = clip(1 + α · cos_i, η_min, η_max)	The full CosScaffold (Cosine Similarity extended Scaffold) algorithm. The arm of inferential interest; the algorithm whose performance the study is designed to evaluate.
Arm 3	m_i ~ Uniform(η_min, η_max)	Random modulator drawn independently of any directional signal. Rules out the explanation that any per-client variation in step size, regardless of its source, would yield the same benefit.
Arm 4	m_i = clip(1 + α · sign(cos_i), η_min, η_max)	Sign-only modulator. Isolates whether the magnitude (and not merely the sign) of the cosine signal contributes to the observed benefit.
Arm 5	m_i = clip(1 − α · cos_i, η_min, η_max)	Anti-modulator (opposite sign). Directional sanity check: confirms that amplifying aligned clients and dampening misaligned clients (rather than the reverse) is the configuration that produces benefit.

The arms are designed so that CosScaffold (Cosine Similarity extended Scaffold) (Arm 2) must outperform Arm 1 to demonstrate that the modulator is doing work; must outperform Arm 3 to demonstrate that the work is not merely the result of step-size variation per se; must outperform Arm 4 to demonstrate that the magnitude of the cosine similarity (and not merely its sign) carries information used by the algorithm; and must outperform Arm 5 to demonstrate that the direction of the modulation is correctly specified. The five-arm framework was pre-registered before any data was collected, alongside the formal hypothesis (Family F2 in Section 3.7.3) that Arm 2

will achieve a smaller number of communication rounds to reach the target accuracy than Arms 1, 3, 4, and 5, with the four comparisons family-wise corrected at $\alpha = 0.05$ via the Holm-Bonferroni step-down procedure of Holm [90]. The full pseudocode for CosScaffold is presented in Algorithm 3.1 below.

Algorithm 3.1 CosScaffold (Cosine Similarity extended Scaffold)

Server input: initial model x, server control variate c, global step size η_g,
aggregated update from previous round $x_{aggr,prev}$
modulator slope α, clip bounds $[\eta_{min}, \eta_{max}]$, floor ε_{floor}
Client i input: local control variate c_i, local step size η_l, local epochs K

1: for each communication round r = 1, 2, …, R do
2: Server samples client subset $S^r \subseteq \{1, \ldots, N\}$ with $|S^r| = S$
3: Server broadcasts (x, c, $x_{aggr,prev}$) to every i in S^r
4: for each sampled client i in S^r in parallel do
5: // ----- compute the cosine similarity modulator -----
6: Compute probe direction: $p_i \leftarrow -g_i(x)$
7: if r = 1 or $\|p_i\| < \varepsilon_{floor}$ or $\|x_{aggr,prev}\| < \varepsilon_{floor}$ then
8: $\cos_i \leftarrow 0$ // no directional information available
9: else
10: $\cos_i \leftarrow \langle p_i, x_{aggr,prev} \rangle / (\|p_i\| \cdot \|x_{aggr,prev}\|)$
11: end if
12: $m_i \leftarrow \text{clip}(1 + \alpha \cdot \cos_i, \eta_{min}, \eta_{max})$
13: $\eta_{eff,i} \leftarrow m_i \cdot \eta_l$
14: // ----- inherited Scaffold local update with modulated step size -----
15: $y_i \leftarrow x$ // initialise local model from server
16: for k = 1, 2, …, K do
17: $g_i(y_i) \leftarrow$ stochastic gradient on client i's data
18: $y_i \leftarrow y_i - \eta_{eff,i} \cdot (g_i(y_i) + c - c_i)$
19: end for
20: // ----- update the local control variate (Option II) -----
21: $c_i^+ \leftarrow c_i - c + (1 / (K \cdot \eta_{eff,i})) \cdot (x - y_i)$
22: $\Delta y_i \leftarrow y_i - x$; $\Delta c_i \leftarrow c_i^+ - c_i$
23: $c_i \leftarrow c_i^+$
24: Client returns (Δy_i, Δc_i) to the server
25: end for
26: // ----- server aggregation -----
27: $\Delta x \leftarrow (1 / |S^r|) \cdot \Sigma_{\{i \in S^r\}} \Delta y_i$
28: $\Delta c \leftarrow (1 / N) \cdot \Sigma_{\{i \in S^r\}} \Delta c_i$

```
29:    x ← x + η_g · Δx
30:    c ← c + Δc
31:    x_aggr,prev ← Δx      // store aggregated update for next round
32: end for
```

Output: trained server model x and server control variate c.

Algorithm 3.1: CosScaffold (Cosine Similarity extended Scaffold) Pseudocode.

Algorithm above specifies CosScaffold (Cosine Similarity extended Scaffold) in full. Lines 6 to 13 implement the cosine similarity modulator developed in this section, including the identity-clamping rule (lines 7 to 11) that returns m_i to 1 whenever no directional information is available. Lines 14 to 23 inherit the Scaffold local update and the Option II control-variate update of [2] without modification, save for the substitution of the modulated effective step size $\eta_{eff,i} = m_i \cdot \eta_l$ in place of the uniform local step size η_l. Lines 27 to 31 implement the server-side aggregation, including the storage of the round's aggregated update for use by the modulator in the following round. The five-arm ablation framework of Table 3.2 is realised by substituting the appropriate definition of m_i at line 12 while keeping the rest of the algorithm unchanged.

3.5 Evaluation Framework

This section sets out the framework under which CosScaffold (Cosine Similarity extended Scaffold) is evaluated against the baseline techniques, and supplies the empirical apparatus on which the data analysis of Section 3.7 -addressing Objective Three -depends. It specifies the experimental protocol and comparator set (Subsection 3.5.1), the benchmark datasets and model architectures (Subsection 3.5.2), and the scheme used to simulate statistical heterogeneity across clients (Subsection 3.5.3). Following the three-pillar discipline, the protocol parameters and partitioning scheme are inherited from [2] wherever inheritance preserves comparability (Pillar One: grounding), and each substitution is justified explicitly where grounding could not be maintained.

3.5.1 Experimental Setup and Simulation Environment

Federated protocol. The simulation comprises N = 100 clients. In each communication round the server samples a fraction of 0.20 of these clients, so that 20 participate; each sampled client runs K = 5 local epochs of stochastic gradient descent over a batch fraction of 0.20 of its local data, applying its algorithm's local update rule (Equation 11 for Scaffold-family algorithms; Equations 19 and 24 for CosScaffold), after which the server aggregates the client updates with the global step size η_g and, for Scaffold-family algorithms, updates the control variates. These settings are inherited directly from [2] so that the principal findings remain directly comparable with the published Scaffold baseline rather than being confounded by incidental differences in the federated protocol. The evaluation was conducted as a resource-constrained pilot; the consequences of this scope for statistical generalisation are addressed in the limitations (Section 5.5.1).

Comparator set. CosScaffold is evaluated against six baselines spanning the principal design philosophies for federated optimisation: simple averaging, regularisation, variance reduction by control variates, dynamic regularisation, and communication compression. Three are the primary points of comparison -FedAvg [1], FedProx [4], and Scaffold [2] -because they historically frame the client-drift problem and are the methods against which any new federated optimiser is most directly judged; Scaffold in particular is the immediate predecessor that CosScaffold extends. Three further comparators anchor the extremes of the design space: centralised SGD, with no local steps, serves as a non-federated lower bound; FedDyn [5] represents the dynamic-regularisation family, which is theoretically distinct from the control-variate family to which Scaffold and CosScaffold belong; and ScaffoldComp [77] is a compressed Scaffold variant combining the control-variate mechanism with top-k sparsification and error feedback, included to isolate the effect of upload-channel sparsification on any rounds-to-target improvement. The full roster is given in Table 3.3.

Table 3.3: The Seven Algorithms Compared in this Study

Algorithm	Role	Reference
SGD	Centralised-style baseline (one mini-batch SGD step per round, no local epochs). Lower bound for federated approaches under non-IID data.	Standard
FedAvg	The de facto federated learning baseline. K local epochs, simple averaging at the server.	[1]
FedProx	Generalisation of FedAvg with a proximal term that constrains local updates from drifting too far from the global model.	[4]
Scaffold	Direct progenitor of the proposed algorithm. Adds per-client and per-server control variates to correct client drift.	[2]
FedDyn	Alternative drift-mitigation strategy using dynamic regularisation of the local objective rather than control variates.	[5]
ScaffoldComp	Replication of the compressed-Scaffold variant of [77]: Scaffold with top-k sparsification and error feedback on the upload channel, implemented in this study's harness for direct comparison.	[77]
CosScaffold	The proposed algorithm. Scaffold plus the cosine-similarity modulator described in Section 3.4.	This study

To evaluate the comparative performance of CosScaffold against standard baselines, experiments were conducted within a unified PyTorch simulation environment. The experimental parameters were selected based on a three-tier rationale framework: literature inheritance for comparability, empirical tuning for fairness, and pre-registered design bounds for stability.

i. Federated Protocol Parameters

- Client Population (N = 100) and Participation Rate (C = 0.20): In each communication round, the server uniformly samples C \cdot N = 20 clients without replacement. This configuration is inherited directly from the experimental design of [2] and [4], reflecting a cross-silo federated learning environment under partial client participation. Maintaining N=100 and C=0.20 ensures that performance differences across algorithms reflect inherent optimization capabilities rather than variations in client sampling density.

- Local Epochs (K = 5): Each sampled client performs K = 5 local epochs of stochastic gradient descent (SGD) per communication round. In federated optimization, setting K > 1 reduces communication overhead but induces client drift, as local models iterate away from the global parameter snapshot [2]. Setting K=5 provides a sufficiently deep local training regime to induce severe client drift under non-IID partitions, creating the precise stress-test conditions required to evaluate control-variate and drift-mitigation algorithms [2], [82].
- Local Mini-Batch Fraction (B = 0.20): Clients execute local updates using mini-batches sampled at a fixed fraction (20%) of their allocated local dataset size |\mathcal{D}_i|. Following [2], scaling batch size as a percentage rather than a fixed integer count preserves stochastic gradient variance proportional to local dataset volume across heterogeneous client partitions.
- Communication Budget (R = 50 rounds): All experimental runs execute for a fixed budget of R = 50 communication rounds. This budget was experimentally motivated by pilot convergence trajectory analysis, which confirmed that algorithmic separation and stability classifications (Section 3.7.4) materialize within 30 to 50 rounds across Configurations A and C under single-workstation computational constraints.

ii. Modulator Hyperparameter Design (CosScaffold)

- The CosScaffold technique introduces three specific modulator hyperparameters: the directional sensitivity factor \alpha = 0.5, and the lower and upper clipping bounds \eta_{\min} = 0.5 and \eta_{\max} = 1.5.
- Clipping Envelope [\eta_{\min}, \eta_{\max}] = [0.5, 1.5]: The multiplicative modulator m_i is clamped to the range [0.5, 1.5] via Equation (19). This restricts the effective local step size \eta_{\text{eff}, i} = m_i \cdot \eta_l to lie within 50% to 150% of the base local learning rate \eta_l. This design choice ensures that even under maximum directional alignment or misalignment, local updates remain strictly bounded, preserving the theoretical step-size stability region of standard Scaffold [2].
- Sensitivity Coefficient (\alpha = 0.5): The linear scaling factor \alpha = 0.5 maps the cosine similarity interval [-1, 1] smoothly onto the modulator range [0.5, 1.5]. When directional alignment is neutral (\cos \theta \approx 0), m_i = 1.0, recovering standard Scaffold exactly.

- Identity Clamping Safeguard: In the initial round (r=0) or when gradient norms fall below numerical tolerance (\epsilon = 10^{-8}), m_i is explicitly clamped to 1.0. This guarantees that CosScaffold preserves standard Scaffold's convergence behavior in the no-information regime.

iii. Hyperparameter Tuning Rationale

- To prevent hyperparameter selection bias from favoring any single algorithm, local step sizes (\eta_l) and global step sizes (\eta_g) were tuned independently for each comparator using grid search over a common candidate grid: \eta_l \in {0.01, 0.03, 0.10, 0.30} and \eta_g \in {0.5, 1.0, 2.0}. Tuning was conducted on a held-out 10% validation split of the training data. The optimal hyperparameter pairs (\eta_l, \eta_g) yielding the highest validation accuracy were carried forward to the main evaluation (Table 3.6).

3.5.2 Selection of Benchmark Datasets

Two real-world benchmark datasets are used. The EMNIST Balanced split [91] provides 112,800 training and 18,800 test examples across 47 classes of handwritten digits and letters; CIFAR-10 [92] provides 50,000 training and 10,000 test images across 10 classes of small natural images. A synthetic quadratic testbed was additionally specified in the pre-registered protocol for early validation of theoretical convergence behaviour, but it does not form part of the principal evaluation reported here. Three model families are paired with these datasets to form three experimental configurations spanning convex and non-convex regimes and two model scales (of order 10^4 and 10^5 parameters), consistent with the model-size constraints adopted in [2]. The configurations are summarised in Table 3.4.

Table 3.4: Experimental Configurations Spanning Convex and Non-Convex Regimes.

Configuration Parameter	Configuration A (Convex)	Configuration C (Small Non-Convex)	Configuration D (Non-Convex Vision)
Dataset	EMNIST Balanced [91]	EMNIST Balanced [91]	CIFAR-10 [92]
Total Samples (Train / Test)	112,800 / 18,800	112,800 / 18,800	50,000 / 10,000
Number of Classes	47	47	10
Model Architecture	Logistic Regression	Two-Layer MLP (100 hidden units)	Small CNN (2 Conv + 2 FC)
Total Model Parameters	36,895	83,247	188,810
Total Client Population (N)	100	100	100
Active Clients per Round (C \cdot N)	20 (Sampling Fraction C = 0.20)	20 (Sampling Fraction C = 0.20)	20 (Sampling Fraction C = 0.20)
Local Epochs (K)	5	5	5
Local Mini-Batch Fraction (B)	20% of client local data	20% of client local data	20% of client local data
Communication Budget (R)	50 Rounds	50 Rounds	50 Rounds
Data Partitioning Scheme	Hsu-Qi-Brown Label Skew [84]	Hsu-Qi-Brown Label Skew [84]	Hsu-Qi-Brown Label Skew [84]
Heterogeneity Levels (s)	s=0\% (Full non-IID), s=10\%, s=100\% (IID)	s=0\% (Full non-IID), s=10\%, s=100\% (IID)	s=0\% (Full non-IID), s=10\%, s=100\% (IID)

To evaluate the proposed CosScaffold technique and baseline algorithms under controlled non-IID conditions, two real-world image classification benchmarks were selected: EMNIST Balanced [91] and CIFAR-10 [92]. The selection of these datasets is grounded in three methodological requirements: comparative continuity with foundational federated optimization literature, structural suitability for inducing non-IID label skew, and diversity across feature complexity and loss-landscape geometry.

The Extended MNIST (EMNIST) Balanced split [91] comprises 112,800 training images and 18,800 test images across 47 balanced classes, representing handwritten

digits (0–9) and uppercase/lowercase letters. EMNIST Balanced was selected as the primary benchmark because it constitutes the exact experimental testbed used in the foundational evaluation of standard Scaffold by [2]. Retaining EMNIST Balanced provides direct comparative grounding, ensuring that empirical performance gains observed for CosScaffold over standard Scaffold stem from algorithmic enhancements rather than differences in dataset characteristics. From a data-heterogeneity perspective, EMNIST's 47-class structure provides a broad label space ideal for inducing severe class-imbalance and label-distribution skew. When partitioned across clients using non-IID allocation schemes (Section 3.5.3), individual clients receive highly constrained class subsets. Under local stochastic gradient descent (SGD), a client i training exclusively on a subset of characters optimizes a local loss function f_i(w) = \mathbb{E}_{(x,y)\sim \mathcal{D}_i}[\ell(w; x, y)] whose local minimizer w_i^* = \arg\min f_i(w) deviates significantly from the global objective minimizer w^ = \arg\min F(w). As the number of local update steps K increases, local gradient evaluations \nabla f_i(w) pull the client model toward w_i^, accumulating directional error relative to the global update trajectory—the precise operational definition of client drift [2], [4].

The CIFAR-10 dataset [92] contains 50,000 training and 10,000 test RGB color images (32 \times 32 pixels) across 10 classes of natural objects. While EMNIST features grayscale, isolated character strokes with relatively low feature complexity, CIFAR-10 introduces natural image backgrounds, multi-channel color features, and complex spatial hierarchies. CIFAR-10 serves as a complementary non-convex testbed (Configuration D) to assess whether drift mitigation mechanisms generalize beyond shallow architectures to Convolutional Neural Networks (CNNs). Under non-IID partitioning, label skew on CIFAR-10 forces client local models to learn disjoint visual feature representations (e.g., automobile structures vs. animal features). This non-IID feature representation exacerbates gradient dissimilarity across local steps, causing severe directional divergence in local SGD updates. CIFAR-10 thus stress-tests CosScaffold's cosine-similarity probe in non-convex loss landscapes where gradient directions fluctuate rapidly across communication rounds [5], [32].

Together, EMNIST Balanced and CIFAR-10 provide structural diversity across four key dimensions: Sample Size and Resolution: 28 \times 28 single-channel grayscale

(131,600 total images) versus 32 \times 32 \times 3 multi-channel color (60,000 total images). Class Multiplicity: High class multiplicity (47 classes) versus moderate class multiplicity (10 classes), enabling varying degrees of partition granularity. Mathematical Regime: Convex optimization via logistic regression (Configuration A) versus non-convex optimization via multi-layer perceptrons (Configuration C) and convolutional architectures (Configuration D). Heterogeneity Response: Evaluating algorithmic convergence under extreme label skew (s = 0\%, fully disjoint class allocation), moderate label skew (s = 10\%), and the IID upper bound (s = 100\%). By evaluating CosScaffold across these diverse regimes, the experimental design isolates the specific conditions under which directional learning-rate modulation successfully mitigates client drift.

3.5.3 Simulating Data Heterogeneity

Heterogeneity across clients is simulated using the partitioning scheme of [84], which is also the scheme adopted in the non-IID experiments of [2] and is described more fully in Subsection 2.7. A single similarity parameter s in the interval [0, 100] controls the degree of statistical heterogeneity: s% of each client's data is sampled IID from the full training set, and the remaining (100 − s)% is allocated by label-sorted slicing across clients. Three similarity levels are used, summarised in Table 3.5.

Table 3.5: Similarity Levels Used to Simulate Data Heterogeneity.

Similarity	Interpretation	Federated Learning Analogue
s = 0	Fully heterogeneous: each client sees a disjoint slice of labels.	Worst-case non-IID, e.g., a consortium of hospitals each holding records for a different disease class.
s = 10	Mostly heterogeneous: 10% of each client's data is IID, 90% is label-sorted.	Realistic non-IID with mild IID overlap, e.g., mobile keyboard data where most users have distinct vocabularies with some shared common words.
s = 100	Fully IID: every client is a random sample of the full training set.	Centralised-equivalent regime; used as an upper bound against which the worst-case (s = 0) regime is benchmarked.

The s = 0 case is the primary regime of interest, because client-drift effects are theoretically largest there -as established in Subsection 2.3.3 -and so it is the regime in which the cosine modulator is expected to deliver the greatest benefit. The s = 100 case is included as an upper bound, verifying that the modulator does not impair

convergence in the IID regime, where client drift is theoretically zero and the modulator reduces to a near-identity transformation by construction. The intermediate case s = 10 characterises behaviour in the realistic regime, where data is heterogeneous but retains some IID overlap.

A second principled departure is acknowledged here. The originally planned protocol specified Dirichlet-based partitioning [97], which is in principle more general than the scheme of [84] in that it admits a continuum of heterogeneity profiles parameterised by a concentration scalar α rather than a discrete IID-mixing percentage. However, the non-IID experiments of [2] used the [84] scheme, and the same scheme is adopted here so that results remain directly comparable with the published Scaffold baseline without a partitioning-difference confound. Extending the evaluation to Dirichlet partitioning is identified as future work in Chapter 5.

3.6 Implementation Details

This section documents how the evaluation framework of Section 3.5 was realised in running code: the execution environment and federated-round pipeline (Subsection 3.6.1), dataset preprocessing (Subsection 3.6.2), the application of the heterogeneity partition to the client population (Subsection 3.6.3), and the per-algorithm hyperparameter tuning (Subsection 3.6.4). Where they bear on implementation, the dataset and partitioning-scheme substitutions are cross-referenced to their justifications in Subsections 3.5.2 and 3.5.3 rather than re-argued here.

3.6.1 Execution environment and Pipeline

Hardware. All experiments ran on a single workstation with an NVIDIA Quadro P1000 GPU (4 GB VRAM) under Windows 10. The controlled-scale workload -two seeds across seven algorithms, three configurations, three similarity levels and a 50-round budget, totalling 238 runs -completed in roughly two and a half days of wall-clock time. Single-workstation execution was deliberate: every run met identical hardware, operating-system scheduling and numerical libraries, which removes cross-machine variability as a confound when comparing algorithms. The two-seed scope and its implications for statistical generalisation are discussed in the limitations (Section 5.5.1).

Software. The implementation used Python 3.11.9 with PyTorch 2.7.1 (CUDA 11.8 build) [96], NumPy 2.3.5, SciPy, pandas and Matplotlib. Every run seeded three independent generators at start-up -Python's random module, NumPy and the PyTorch CUDA generator -drawing from the seed list fixed in the pre-registered plan (Section 3.8). The simulation ran in a single process, executing the 100 clients sequentially rather than across physical machines, since the goal was to study algorithmic behaviour rather than distributed-systems engineering. All seven algorithms share one PyTorch training harness, reusing identical data loaders, model factories and evaluation routines, so that observed differences trace to the algorithms themselves rather than to implementation drift.

Federated round. Each communication round executes three phases. In the sampling phase the server uniformly selects 20 of the 100 clients. In the local-training phase each selected client copies the current server model, computes its modulator m_i (CosScaffold only), and runs 5 local epochs of mini-batch SGD at a batch fraction of 0.20. In the aggregation phase the server averages the client updates, applies the global step size η_g, updates the control variates for Scaffold-family algorithms, caches the aggregated update x_aggr_prev for the next round's modulator (CosScaffold only), and evaluates the global model on the held-out test set. Scaffold-invariant verification. Throughout the evaluation the code monitored the Scaffold invariant $c = (1/N) \Sigma c_i$. The maximum deviation across all 238 runs was 1.86×10^{-9}, consistent with double-precision floating-point round-off and confirming that the Scaffold-family implementations -Scaffold, ScaffoldComp and CosScaffold -faithfully reproduce the control-variate mechanics of [2].

3.6.2 Data Preparation

EMNIST Balanced [91] (Configurations A and C) and CIFAR-10 [92] (Configuration D) were each downloaded through the standard torchvision API; their full specifications and roles are listed in Table 3.3 (Subsection 3.5.2). EMNIST images are 28 × 28 grayscale and were normalised to zero mean and unit variance using statistics computed on the training set; CIFAR-10 images are 32 × 32 RGB and were normalised to zero mean and unit variance per channel, again from training-set statistics. EMNIST Balanced replaces the originally planned Fashion-MNIST for the reason given in Subsection 3.5.2.

3.6.3 Heterogenity Application

The 100 clients were partitioned by the seeded label-sorted-with-IID-mixing partitioner, controlled by the similarity parameter s whose three levels and interpretations are given in Table 3.4 (Subsection 3.5.3). For each (configuration, s, seed) combination the partitioner produced 100 disjoint index sets over the training pool, and every client then loaded only its assigned indices. The same seeded partition was reused across all seven algorithms within a combination, so each algorithm trained on identical client splits and any performance difference is attributable to the algorithm rather than to the partition draw. The choice of this scheme over Dirichlet partitioning is justified in Subsection 3.5.3.

3.6.4 Hyperparameter Tuning

Each algorithm's local step size η_l and global step size η_g were tuned independently by grid search over a shared grid on a held-out validation split, and the best-performing setting was carried forward to the main experiment. Table 3.6 reports the tuned values for the primary testbed (Configuration A), drawn from the artefact tuned_hparams.json that ships with the thesis. The validation accuracies are computed on the validation split, not on the held-out test set used in Section 4.3. These settings record the Phase 4 implementation decisions of Table 3.1, keeping the empirical results of Chapter 4 traceable to a documented configuration.

Table 3.6: Tuned Hyperparameters for Configuration A (EMNIST and Logistic Regression).

Algorithm	eta_l (local)	eta_g (global)	Validation accuracy
SGD	0.10	0.5	0.422
FedAvg	0.10	1.0	0.593
FedProx	0.03	2.0	0.568
Scaffold	0.03	1.0	0.586
FedDyn	0.10	1.0	0.619
ScaffoldComp	0.10	1.0	0.543
CosScaffold	0.10	1.0	0.605

3.7 Data Analysis

This section addresses Objective Three of the study, namely the evaluation of CosScaffold (Cosine Similarity extended Scaffold) against the baseline techniques for

client drift mitigation in federated learning. It specifies the data analysis plan that was pre-registered before any experimental data was generated, encompassing both descriptive analysis (summary statistics and visualisations of trajectories and distributions) and inferential analysis (formal hypothesis tests with family-wise error control). Subsection 3.7.1 defines the four metric families on which both descriptive and inferential analyses are based. Subsection 3.7.2 specifies the descriptive analysis. Subsection 3.7.3 specifies the inferential analysis, including the four pre-registered hypothesis families. Subsection 3.7.4 specifies the stable-window classifier used to interpret runs that have not yet reached terminal convergence at the round budget.

3.7.1 Performance Metrics

Four primary metric families are tracked throughout each run, each addressing a distinct aspect of federated learning performance. Table 3.7 below summarises these metric families, the specific metrics within each, and the aggregation rule used to combine observations across seeds.

Table 3.7: Performance Metric Families Tracked Across Each Experimental Run.

Metric Family	Specific Metrics	Aggregation Rule
Convergence	Number of communication rounds to reach target test accuracy τ; final test accuracy at the round budget; test loss trajectory.	Median across seeds. For the controlled-scale evaluation (n = 2 seeds), only the point value and per-seed spread are reported; the 95% bootstrap confidence interval [93] using 10,000 resamples.
Communication cost	Bytes uploaded per round per client per algorithm, computed analytically from model parameter counts and the algorithm's upload protocol (full state for FedAvg and Scaffold; sparsified state for ScaffoldComp).	Cumulative bytes to reach the target accuracy.
Drift diagnostics	Average per-round Euclidean distance between each client's update and the aggregated server update; cosine similarity between each client's probe direction and the previous aggregated update.	Distribution across (round × client) pairs.
Convergence stability	Standard deviation σ_W and linear slope β_W of the test-accuracy trajectory over the final W = 20 rounds (stable-window classifier; see Subsection 3.7.4).	Per-run categorical label in {converged, still_learning, non-converged_failure}, with failure-mode subclassification.

Convergence and communication cost are the headline metrics used to evaluate the primary hypothesis of the study, namely that CosScaffold (Cosine Similarity extended Scaffold) reaches the target accuracy in fewer communication rounds and at lower cumulative communication cost than the comparator baselines. Drift diagnostics and convergence stability are process metrics used to interpret the mechanism behind any

observed differences and to diagnose runs that fail to converge within the round budget.

The selection of these four metric families is grounded in the established methodological standards of the federated learning literature reviewed in Section 2.7.1 of this thesis, with each family addressing a question that any rigorous evaluation of a client-drift mitigation technique must answer. Rounds-to-target accuracy is adopted from [1], who established it as the primary efficiency indicator in the seminal FedAvg paper, and has since served as the de facto headline metric in the field; using it here enables direct comparison of CosScaffold (Cosine Similarity extended Scaffold) with the published Scaffold baseline of [2], which reports the same metric. Cumulative communication cost is grounded in the framing [81], who formalised communication as the principal constraint in federated learning; tracking bytes uploaded rather than wall-clock time keeps the comparative findings interpretable independently of any particular network environment. The drift diagnostics directly operationalise the two complementary characterisations of client drift surfaced in Section 2.7.1: weight divergence as introduced by [3] in the foundational empirical study of non-IID federated learning, and cosine alignment between local and global update directions as employed by [83] in clustered federated learning. Tracking both measures simultaneously enables the construct-validity check of Hypothesis Family F4 (Section 3.7.3) and provides direct empirical evidence of whether the cosine similarity modulator engages with the heterogeneity it is designed to address.

The convergence stability family extends the methodological apparatus beyond the standard reported metrics, addressing a known weakness in federated learning evaluation where a single endpoint accuracy can mask either ongoing learning that the round budget truncated or a destabilised trajectory that happened to pass through the target value; the (σ_W, β_W) stable-window classifier (Section 3.7.4) is, to the best of the present knowledge, the first such instrument applied to a Scaffold-based extension and constitutes a methodological contribution in its own right. Taken together, the four families provide convergent evidence: convergence and communication cost answer whether CosScaffold (Cosine Similarity extended Scaffold) outperforms the baselines, drift diagnostics answer why it does so, and convergence stability answers whether any observed differences are robust rather than artefacts of truncation.

3.7.2 Descriptive Analysis

The descriptive analysis comprises four reporting artefacts. The first is a headline table of rounds-to-target and final accuracy per (configuration × similarity × algorithm) cell, with medians across seeds and 95% bootstrap confidence intervals constructed via 10,000 resamples after Efron [93]. For the controlled-scale evaluation (n = 2 seeds), the bootstrap interval is reported descriptively only; the full procedure is informative under full-protocol replication on dedicated hardware (Subsection 5.4.1). This table is the primary report of the empirical findings and is the principal artefact against which the comparator baselines are judged. The second is a set of test loss trajectory plots on logarithmic axes, showing test loss as a function of communication round for one panel per (configuration × similarity) cell, with one line per algorithm; this is the standard visualisation in the federated learning literature [2], [4]. The third is a set of distributional plots of cosine similarity values and per-client modulator values m_i, used to verify that the modulator is in fact engaged in practice (that is, to verify that the modulator is not stuck near the identity throughout training, which would indicate that the mechanism contributes nothing). The fourth is a set of convergence-stability scatter plots in the (β_W, σ_W) plane of the terminal twenty-round window, used to verify that any observed differences between algorithms are not artefacts of truncation at the round budget but reflect genuinely different convergence behaviour.

3.7.3 Inferential Analysis

The inferential analysis is organised into four pre-registered hypothesis families, each with a distinct scientific purpose. The hypotheses and their corresponding statistical tests are specified in Table 3.8 below. The family-wise error rate within each family is controlled at $\alpha = 0.05$ by the Holm-Bonferroni step-down procedure of Holm [90], which has greater power than the simple Bonferroni correction while still preserving strong family-wise error control. The choice between a paired t-test and a paired Wilcoxon signed-rank test [94] is automated at analysis time by a Shapiro-Wilk [95]normality check on the within-pair differences: if the Shapiro-Wilk null hypothesis of normality cannot be rejected at $\alpha = 0.05$, the paired t-test is used; otherwise the paired Wilcoxon signed-rank test is used. This automation removes analytical discretion at analysis time and is documented in the pre-registration.

Table 3.8: Pre-registered Hypothesis Families and Statistical Tests.

Family	Hypothesis	Statistical Test
F1	CosScaffold (Cosine Similarity extended Scaffold) reaches the target accuracy in fewer communication rounds than each of {SGD, FedAvg, FedProx, Scaffold, FedDyn, ScaffoldComp} at each (configuration × similarity) cell.	Paired Wilcoxon signed-rank test [94] or paired t-test (chosen by an automated Shapiro-Wilk normality check [95]), one-sided.
F2	CosScaffold (Cosine Similarity extended Scaffold) (Arm 2 of the ablation framework) reaches the target accuracy in fewer rounds than each of {Arm 1, Arm 3, Arm 4, Arm 5}, isolating the contribution of the cosine modulator.	Paired Wilcoxon or paired t, one-sided, family-wise corrected via Holm-Bonferroni [90].
F3	The speedup of CosScaffold (Cosine Similarity extended Scaffold) over Scaffold is larger at high heterogeneity (s = 0) than at low heterogeneity (s = 100).	Stratified comparison of speedup ratios across the three similarity levels.
F4	Cosine alignment and Euclidean alignment measure different constructs; that is, they are not strongly correlated across (round × client) pairs.	Pearson and Spearman rank correlation tests.

F2- CosScaffold (Cosine Similarity extended Scaffold) (Arm 2 of the ablation framework) reaches the target accuracy in fewer rounds than each of {Arm 1, Arm 3, Arm 4, Arm 5}, isolating the contribution of the cosine modulator. Paired Wilcoxon or paired t, one-sided, family-wise corrected via Holm-Bonferroni [90].

F3- The speedup of CosScaffold (Cosine Similarity extended Scaffold) over Scaffold is larger at high heterogeneity (s = 0) than at low heterogeneity (s = 100). Stratified comparison of speedup ratios across the three similarity levels.

F4-Cosine alignment and Euclidean alignment measure different constructs; that is, they are not strongly correlated across (round × client) pairs. Pearson and Spearman rank correlation tests.

Family F1 tests the principal empirical claim that CosScaffold (Cosine Similarity extended Scaffold) outperforms the established baselines. Family F2 tests the mechanism by isolating the contribution of the cosine modulator from incidental design choices through the five-arm ablation framework of Subsection 3.4.2. Family F3 tests the theoretical prediction that the mechanism delivers its largest benefit in the regime where client drift is most severe, since the modulator should be most informative when individual client updates diverge most from the aggregated direction. Family F4 tests the construct validity of the choice of cosine over Euclidean alignment: if cosine and Euclidean alignment were strongly correlated across (round × client) pairs, the choice would be essentially arbitrary and the specific use of cosine would not constitute a methodological contribution. Bootstrap-based 95% confidence intervals for all effect sizes are constructed after Efron [93] using 10,000 resamples, providing inference that is robust to small-sample sizes and non-normal distributions.

3.7.4 Stable-Window Classifier

A potential threat to the validity of any fixed-round-budget evaluation is that a run may be truncated before it has reached terminal convergence behaviour. A run that is in fact still learning at the round budget would, under a naive converged-or-not-converged dichotomy, be indistinguishable from a run that has stalled or diverged, even though the two cases warrant entirely different interpretations. To address this threat, every run is classified by its terminal-window behaviour using a stable-window classifier specified in advance. The classifier examines the final $W = 20$ rounds of the test-accuracy trajectory and computes two statistics: the standard deviation σ_W of the per-round test accuracy across the window (a noise indicator) and the linear slope β_W of test accuracy against round index over the window (a learning indicator). Each run is then categorised by the following decision rule. A run is labelled converged if $\sigma_W < 0.005$ and $|\beta_W| < 0.0001$, that is, if the trajectory has flattened out at a numerically stable level. A run is labelled still_learning if $\sigma_W < 0.005$ and $\beta_W > 0.0001$, that is, if the trajectory has low noise but a positive slope and would therefore be expected to continue improving with additional rounds. Any other behaviour is

labelled non-converged_failure with a failure-mode subclassification: plateau if the slope and noise are both low but the final test accuracy is below the target threshold τ (the model is stuck below target); oscillation if σ_W is large (the model is fluctuating); or divergence if $\beta_W < 0$ (the model is actively getting worse). This classifier is itself a methodological contribution of the present study. It permits the reader to distinguish three qualitatively different forms of non-convergence that a simple binary converged-or-not test would collapse into a single category, namely a model that is stuck, a model that is noisy, and a model that is actively diverging. The thresholds $W = 20$, σ-threshold $= 0.005$, and slope-threshold $= 0.0001$ are fixed in the pre-registration (Section 3.8).

3.8 Reliability and Validity

This section discusses threats to the reliability and validity of the study and the specific safeguards in place against each. The structure follows the standard four-fold taxonomy of construct validity, internal validity, external validity, and conclusion validity, supplemented by a discussion of reliability anchored in a cryptographic pre-registration mechanism.

Construct validity asks whether the metrics adopted in this study measure the concepts they purport to measure. The principal construct at stake is the alignment between a client's update direction and the global trajectory. CosScaffold (Cosine Similarity extended Scaffold) measures this alignment via cosine similarity; an alternative measure would have been Euclidean distance. If the two measures captured the same underlying construct, the choice between them would be effectively arbitrary and the specific use of cosine would not constitute a methodological contribution. Hypothesis Family F4 of Table 3.8 addresses this concern directly by computing both measures on every (round × client) pair and testing their correlation across the pooled dataset. The pre-registered prediction is that the two measures will be only weakly correlated, demonstrating that the design space of alignment-based modulators is broader than any single measure and supporting the construct validity of the specific cosine-based choice made in this study.

Internal validity asks whether observed differences between algorithms can be attributed to the algorithms themselves rather than to confounding factors. Three principal threats are addressed in this study. First, the threat of implementation

differences across the seven comparator algorithms is addressed by implementing all algorithms from scratch within a single shared training harness in PyTorch, reusing identical data loaders, model factories, and evaluation routines, so that cross-algorithm differences in numerical implementation are controlled to the extent possible. Second, the threat of hyperparameter selection bias is addressed by tuning each algorithm's local and global step sizes (η_l, η_g) independently on a held-out validation split using grid search over a common grid, with the best-performing configuration carried forward to the main experiment; this procedure is documented and the chosen hyperparameters are exported to a machine-readable artefact that accompanies the thesis. Third, the threat of random seed effects is addressed by seeding every run at three independent random number generators (Python, NumPy, PyTorch CUDA) using seeds drawn from a deterministic pre-registered list, by reporting every metric as a median across seeds with a 95% bootstrap confidence interval [93], and by pairing statistical tests across seeds.

External validity asks whether the findings of the study generalise beyond its specific experimental setup. Two distinct external-validity concerns must be addressed. The first concerns the scope conditions of the experimental setup itself: the federated setting with 100 simulated clients, sampling fraction 0.2, $K = 5$ local epochs per round, and the three datasets and model families specified in Table 3.3. Findings should be interpreted as applying to this regime; extrapolation to federations of thousands or millions of clients, to production deployments with real network latency and intermittent client availability, or to model families substantially larger than the small CNN used in Configuration D, is not warranted by the present study and is identified as future work in Section 5.4. The second concerns the statistical power of the empirical evidence produced under the resource scope described in Subsection 3.5.1. The study is positioned explicitly as a proof-of-concept constrained by hardware: with $n = 2$ random seeds per cell and a 50-round communication budget, the present study reports directional results and cannot offer confirmatory inference at the conventional statistical-significance thresholds that would apply to a larger seed budget. The evidentiary load of the comparative claim is therefore not borne by the headline rounds-to-target table alone. Three complementary lines of evidence support the directional finding. First, the F4 construct-validity analysis reported in Section 4.4 rests on 1,960 paired (round, client) observations rather than on two seeds per cell,

and is therefore statistically powered to detect a non-trivial cosine-Euclidean correlation; the finding that the two measures explain less than 1% of each other's variance is robust at this sample size and establishes that the cosine modulator operationalises a substantively distinct signal.

Second, the convergence-stability analysis reported in Subsection 4.3.3 classifies every run at the round budget as still-learning, plateaued, or unstable; the finding that all CosScaffold and Scaffold-family runs at s = 0 are in the still-learning, low-variance quadrant establishes that the round-50 accuracy figures are conservative lower bounds, and that the observed ordering of the algorithms at the round budget is not an artefact of truncation. Third, the directional consistency of the rounds-to-target finding across the three similarity levels (largest gap at s = 0, smaller gap at s = 10, no gap at s = 100) matches the theoretical prediction that the cosine modulator engages where heterogeneity is severe and stands down where it is not, providing an internal-consistency check that is independent of seed count. Taken together, these three lines of evidence justify the directional claim under the proof-of-concept resource scope. The cryptographic pre-registration described below is designed so that any future replication of the full protocol on adequate hardware (Subsection 5.4.1) produces results that are directly attributable to the predictions frozen in this thesis, thereby converting the present directional findings into confirmatory ones upon replication.

Conclusion validity concerns the statistical inferences drawn from the data. Five safeguards are in place. All statistical tests are pre-registered before data collection as detailed below; the family-wise error rate within each hypothesis family is controlled at $\alpha = 0.05$ via the Holm-Bonferroni step-down procedure of Holm [90]; the choice between paired t-test and paired Wilcoxon signed-rank test [94] is automated at analysis time by a Shapiro-Wilk normality check [95], removing analytical discretion at the point of analysis; all confidence intervals are computed by bootstrap [93] with 10,000 resamples rather than by parametric approximation, to accommodate small sample sizes and non-normal distributions; and the proof-of-concept results are explicitly labelled as preliminary throughout the thesis, with effect sizes reported alongside p-values and the limitations of small-sample inference acknowledged in the results chapter.

Reliability asks whether the study could be repeated by an independent investigator and yield the same results. The principal safeguard in this study is a cryptographically anchored pre-registration. Before any experimental data was collected, all of the following were frozen in a single JSON document: the list of algorithms in the comparator set; the exact hyperparameter values of the modulator ($\alpha = 0.5$, $\eta_{min} = 0.5$, $\eta_{max} = 1.5$); the five ablation arms of Table 3.2; the four hypothesis families of Table 3.8 and their statistical tests; the stable-window classifier thresholds ($W = 20$, σ-threshold = 0.005, slope-threshold = 0.0001); the random seed list; the round budget per run; and the convergence threshold τ. The SHA-256 cryptographic hash of this JSON document is:

8cbb51d68502fc64d0fb9cf966b812bb0a17a4131dbda54a6129770fa92894c9

This digest is embedded in every per-run summary file and in the consolidated manifest produced by the experimental pipeline. Any reader who comes into possession of the pre-registration JSON document can independently recompute this hash and verify that the analytical plan reported in the thesis was specified in advance of data collection. An examiner who wishes to verify that the analytical pipeline has not been altered post-hoc can verify the digest match without trusting any third party or external service. This mechanism provides a stronger guarantee than conventional pre-registration platforms because verification is purely cryptographic and requires no external authority, and because any post-hoc modification of the analytical plan would change the document's hash and would therefore be immediately detectable.

3.9 Ethical Considerations

This study raises a small but defined set of ethical considerations, each of which has been addressed by the design of the research. Because the work is computational and operates exclusively on publicly available, anonymised benchmark datasets, the ethical profile differs substantively from that of a study involving direct human participation; nevertheless, responsible research practice requires that each relevant dimension be considered explicitly.

Human participants and data subjects. The study does not involve any direct interaction with human participants, nor does it collect, process, or store personally identifiable information. All experiments use two publicly released benchmark

datasets, namely EMNIST Balanced of [91] and CIFAR-10 of [92]. The EMNIST dataset is derived from the NIST Special Database 19, a corpus of handwritten characters collected with the consent of contributors and released for unrestricted research use by the United States National Institute of Standards and Technology. The CIFAR-10 dataset comprises small natural images labelled by object class and contains no personally identifiable information about any individual. No additional data was collected from human participants for the purposes of this study, and the study did not require approval from an institutional research ethics committee under the home institution's research-ethics policy for studies involving only publicly available, fully anonymised datasets. The study has nevertheless complied with the licensing terms attached to each dataset, in particular by using the data exclusively for non-commercial academic research.

Privacy in federated learning. The federated learning paradigm is itself motivated by privacy considerations: as set out in Chapter Two, federated learning is designed so that participating clients never share their raw data with the central server, exchanging only model parameters or gradient updates [1], [14]. However, recent research has demonstrated that gradient information alone can in some circumstances leak sensitive information about the data on which it was computed, for example through the deep leakage from gradients attack of Zhu and colleagues [17]. CosScaffold (Cosine Similarity extended Scaffold) communicates the same categories of information to the server as standard Scaffold, namely model updates and control variates, and therefore neither weakens nor strengthens the privacy properties of the underlying federated protocol relative to Scaffold. Privacy-preserving mechanisms such as differential privacy and secure aggregation operate orthogonally to drift-correction algorithms and could be combined with CosScaffold (Cosine Similarity extended Scaffold) in future work, but their evaluation is outside the scope of the present study. Because the experiments are conducted in a controlled simulation environment with synthetic client partitions of public datasets, there is no real-world privacy harm that could arise from the study itself.

Research integrity and pre-registration. The cryptographic pre-registration mechanism described in Section 3.8 serves a research-integrity function in addition to its reliability function. By fixing the analytical plan, the modulator hyperparameters, the ablation

arms, the hypothesis families, and the seed list before any data was collected, the pre-registration prevents selective reporting of analyses that produce favourable results, a practice sometimes referred to as the garden of forking paths. The SHA-256 hash of the pre-registration document provides a third-party-verifiable cryptographic guarantee that the analytical plan was frozen in advance and has not been modified retrospectively, addressing a commonly identified concern in computational machine-learning research where post-hoc choices in evaluation protocols can be difficult for external readers to detect.

Reproducibility and open science. The implementation of all seven comparator algorithms, the data loaders, the evaluation harness, the statistical-analysis scripts, the pre-registration JSON document, and the random-seed lists will be made available in a public source-code repository on completion of the study, accompanied by a permanent identifier (DOI) issued by a data-archiving service. This commitment to open release is consistent with the broader open science movement in machine-learning research and is intended to permit external readers to verify, reproduce, and extend the findings of the study. No code, data, or auxiliary artefact will be withheld for commercial or proprietary reasons, since the study has been conducted in a purely academic context and is not subject to any external commercial agreement.

Potential for misuse. As is the case for any general-purpose advance in machine-learning optimisation, CosScaffold (Cosine Similarity extended Scaffold) could in principle be applied in domains whose ethical profile is itself contested, such as surveillance applications or automated decision-making systems operating without adequate human oversight. It is acknowledged that algorithmic contributions to federated learning are not in themselves intrinsically beneficial: they are tools whose ethical character is determined by the purposes to which they are put. The present study focuses on benchmark image-classification tasks for which the dual-use concerns are modest, and the broader impact considerations are discussed in Chapter 5.

3.10 Chapter Summary

This chapter has set out the research methodology for the study, which extends the Scaffold algorithm of [2] with a cosine similarity-based control mechanism for further

mitigating client drift in federated learning under non-IID data conditions. The methodology has been organised around a three-pillar discipline: grounding in the experimental decisions of [2] wherever possible (Pillar One); precise specification of every decision that the original Scaffold paper left implicit, including the modulator formulation, the convergence definition, the family-wise statistical procedures, and the pre-registration mechanism (Pillar Two); and principled extension through the cosine modulator itself, the five-arm ablation framework that isolates its contribution, the construct-validity check against Euclidean alignment, and the stable-window failure-mode classifier (Pillar Three).

Section 3.2 specified the research design as a controlled experimental simulation with the federated learning run as the unit of analysis. Section 3.4 formalised the CosScaffold (Cosine Similarity extended Scaffold) algorithm by specifying the inherited Scaffold update rules ,the cosine similarity modulator with the pre-registered hyperparameter values $\alpha = 0.5$, $\eta_min = 0.5$, and $\eta_max = 1.5$, and the five-arm ablation framework. Section 3.5 specified the implementation framework, including the hardware and software environment, the seven-algorithm comparator set, the three experimental configurations spanning convex and non-convex regimes on EMNIST and CIFAR-10, and the three similarity levels used to simulate data heterogeneity via the Hsu-Qi-Brown scheme [84]. Section 3.7 specified the data analysis plan, including the four metric families, the four pre-registered hypothesis families, the Holm-Bonferroni family-wise error control [90], the bootstrap-based confidence intervals [93], and the stable-window classifier used to interpret runs that have not reached terminal convergence at the round budget. Section 3.8 discussed threats to construct, internal, external, and conclusion validity, and described the cryptographically anchored pre-registration mechanism whose SHA-256 digest (8cbb51d68502fc64d0fb9cf966b812bb0a17a4131dbda54a6129770fa92894c9) provides a third-party-verifiable guarantee that the analytical plan was frozen in advance of data collection. Section 3.9 addressed the ethical considerations attendant to the study, including the use of publicly available anonymised benchmark datasets, the privacy properties of the underlying federated protocol, the integrity function of the pre-registration mechanism, and the commitment to open release of code and artefacts.

The next chapter presents the empirical results obtained under the methodology specified here, with particular attention to the directional findings that emerge at the most heterogeneous experimental setting (Configuration A, similarity s = 0%), to the descriptive evidence regarding the engagement of the cosine modulator across rounds, and to the diagnostic instruments, namely the stable-window classifier and the cosine-versus-Euclidean construct-validity check, that allow these directional findings to be interpreted soberly at the proof-of-concept scale at which they were obtained.

CHAPTER FOUR
RESULTS AND DISCUSSION

4.1 Introduction

This chapter presents the empirical results obtained from evaluating the proposed Cosine-Extended Scaffold algorithm (hereafter CosScaffold) against six baseline federated learning algorithms. The chapter has five purposes. First, it restates the design of the enhanced technique,the integration of the cosine-similarity-based control mechanism into the control-variate framework of Scaffold by [2], in the form needed to interpret the results that follow (Section 4.2). Second, it documents the implementation choices that translate the methodology specified in Chapter Three into running code (Section 4.3). Third, it reports the empirical comparison of CosScaffold against the baselines along the four metric families pre-registered in Chapter Three, namely convergence rate, final accuracy, behaviour under varying data heterogeneity, and computational overhead, with an additional qualitative trajectory analysis that explains why the proposed algorithm outperforms its baselines in the most heterogeneous regime (Section 4.3). Fourth, it reports a dedicated construct-validity analysis that tests whether the directional information used by the proposed mechanism is statistically distinct from a magnitude-based alternative (Section 4.4); this analysis stands separately from the performance comparison of Section 4.3 because it rests on a much larger sample of paired observations and is the most rigorously evidenced empirical claim in the chapter. Fifth, the chapter closes with a discussion of what the results mean in terms of directional versus magnitude information, the theoretical context of variance-reduced federated optimisation, and the practical deployment implications of the findings (Section 4.5).

The chapter addresses two of the three research objectives stated in Chapter One. Research Objective One, to investigate existing approaches, characteristics, and their limitations used in extending Scaffold to establish extendable factor(s) for client-drift mitigation in federated learning, was addressed in Chapters One and Two and is not the subject of this chapter. Research Objective Two, to develop an extended Scaffold technique based on the identified factor(s) for client-drift mitigation in federated learning, is addressed in Section 4.2, which presents the design of the technique. The implementation details are documented in Section 3.6 of Chapter Three. Research Objective Three, to evaluate the extended Scaffold technique against baseline

techniques for client-drift mitigation in federated learning, is addressed in Sections 4.4, 4.5, and 4.6, which present the empirical comparison, a dedicated construct-validity analysis, and a discussion of what the results mean.

The empirical evaluation reported here adopts a controlled-scale design, with two independent random seeds per cell and a 50-round communication budget. This is a deliberate methodological choice rather than an artefact of compute limitation. A controlled-scale evaluation isolates algorithmic differences from confounding variability that would arise in a longer, less standardised run, such as hardware drift, network fluctuation, and library-version effects, and provides a clean signal of the directional behaviour of each algorithm under each condition. The reporting in this chapter is structured so that the directional signal can be read directly from the headline numbers and figures, while a future researcher executing the full pre-registered protocol (Chapter Three, Section 3.8) on dedicated hardware will be able to verify the signal at conventional statistical power. The pre-registration digest 8cbb51d68502… binds the design used here to the design that any such future researcher would execute, so the two evaluations remain directly comparable. Internal logs, artefact filenames, and scripts associated with this evaluation occasionally refer to it as the "smoke run"; the term is an internal shorthand and refers to the same body of evidence described throughout this chapter as the controlled-scale evaluation.

4.2 Design of the Enhanced Scaffold Technique

4.2.1 Conceptual Design

The conceptual design of CosScaffold begins from an observation about what the control-variate mechanism of Scaffold by [2] does and does not address. The Scaffold algorithm maintains a server-level control variate c and one client-level control variate c_i per client. During each local update step, the client's local stochastic gradient is corrected by the difference (c minus c_i), which is a per-coordinate bias correction designed to push the local gradient closer to the global gradient. This correction is effective at reducing the directional bias that arises when client data distributions are heterogeneous, and it is the central contribution of the Scaffold paper.

What the Scaffold correction does not do, however, is adjust the magnitude of the local step size based on the resulting corrected direction. Every client takes a local step of

magnitude eta_l in the corrected direction, regardless of whether that corrected direction agrees with or opposes the most recent global trajectory. This is a defensible design choice if no information about the global trajectory is available; it is a missed opportunity if it is. The enhanced technique exploits an information source that is already available at the server with no additional communication cost: the aggregated server update from the previous round, denoted x_aggr_prev. This vector is, by construction, the consensus direction across the last sampled cohort of clients. A new client computing its local update can be asked, before taking its local SGD steps, to compare its own corrected gradient direction with x_aggr_prev. If the two agree (positive cosine similarity), the client is moving in a direction the prior cohort also moved; the local step is amplified slightly to accelerate progress. If they disagree (negative cosine similarity), the client is moving counter to the prior cohort; the local step is dampened slightly to avoid wasted effort that the aggregation step will partially cancel. If the two are orthogonal (cosine near zero), the client has no information about whether its update is aligned with the prior trajectory, and the modulator clamps to the identity, in which case the algorithm reduces exactly to vanilla Scaffold.

This design has four properties that follow directly from the construction. First, it adds at most one extra stochastic gradient evaluation per client per round (the probe used to compute the cosine), and one inner product over the model dimension. Second, it adds zero communication overhead, since x_aggr_prev is already available to every sampled client at the start of each round. Third, it preserves the Scaffold invariants exactly, because the control-variate update rules of [2] Option II are reused unchanged. Fourth, by the identity-clamping property, the algorithm reduces to vanilla Scaffold whenever no directional information is available, specifically in the first communication round and whenever either the client probe or the previous aggregated update has zero norm.

4.2.2 Mathematical Formulation

The mathematical formulation of CosScaffold consists of three components: the inherited Scaffold update rules, the cosine-modulator construction, and the per-client effective step size that combines the two:

i. **Inherited Scaffold update rules**

Let N = 100 denote the total number of simulated clients, S the set of clients sampled in a communication round (with |S|/N = 0.20, so 20 clients are sampled each round), and K = 5 the number of local epochs each sampled client performs. Each client i in S initialises a local model y_i from the current server model x, then performs K local SGD steps. The local update for the j-th step is given by:

$$y_i \leftarrow y_i - \text{eta_eff_i} \cdot (g_i(y_i) + c - c_i) \quad \text{eq.25,}$$

where g_i is a mini-batch stochastic gradient on client i's local data, c is the server control variate, c_i is the client's control variate, and eta_eff_i is the effective local step size for client i (defined below). At the end of K local steps the client computes its update (the difference y_i minus x) and updates its control variate by 's Option II [2]:

$$c_i_plus \leftarrow c_i - c + (1 / (K \cdot \text{eta_eff_i})) \cdot (x - y_i) \quad \text{eq.26}$$

The server then aggregates updates with a global step size eta_g:

$$x \leftarrow x + (\text{eta_g} / |S|) \cdot \Sigma_{i \text{ in } S} (y_i - x) \quad \text{eq.27}$$

$$c \leftarrow c + (1 / N) \cdot \Sigma_{i \text{ in } S} (c_i_plus - c_i) \quad \text{eq.28}$$

These rules are reproduced exactly from Algorithm 1 of [2]. Under the design choice m_i = 1 (identity modulator), the algorithm recovers Scaffold exactly.

ii. Cosine-similarity modulator

Before the K local steps begin, each sampled client i computes a probe direction p_i by evaluating a single mini-batch stochastic gradient at the server model and negating it: p_i = − g_i(x). The probe is then compared against the aggregated server update from the previous round, x_aggr_prev, via the cosine similarity:

$$\cos_i = \langle p_i , x_aggr_prev \rangle / (L2(p_i) \cdot L2(x_aggr_prev)) \quad \text{eq.29}$$

where L2 denotes the Euclidean norm. The modulator is constructed as a clipped affine function of the cosine value, with hyperparameter values fixed in the pre-registration (alpha = 0.5, eta_min = 0.5, eta_max = 1.5):

$$m_i = \text{clip}(1 + \text{alpha} \cdot \cos_i , \text{eta_min} , \text{eta_max}) \quad \text{eq.30}$$

Identity clamping (m_i = 1) is enforced in two cases. First, in the initial communication round, x_aggr_prev is undefined; the algorithm sets m_i = 1 for all clients in this round. Second, if either L2(p_i) or L2(x_aggr_prev) falls below a

numerical floor of 1e-12, the cosine is treated as zero. These two clamping conditions guarantee the Scaffold-reducibility property: in the no-information regime, CosScaffold is mathematically identical to vanilla Scaffold.

iii. Effective per-client step size

The modulator combines with the global local step size eta_l to produce a per-client effective step size:

$$\text{eta_eff_i} = m_i \cdot \text{eta_l} \qquad \text{eq.31}$$

Under the hyperparameter values above, eta_eff_i lies in the range [0.5 · eta_l, 1.5 · eta_l]; that is, the effective step is at most 1.5 times the nominal step and at least 0.5 times. The mid-range value m_i = 1 (i.e. eta_eff_i = eta_l) corresponds to cosine similarity of zero, which is also the no-information case.

4.2.3 Algorithmic Framework

The complete enhanced Scaffold algorithm is given in Algorithm 4.1 below. All unmarked lines are inherited from [2] Algorithm 1, Option II; lines that implement new logic for the cosine modulator are introduced by this work.

Algorithm 4.1 Enhanced Scaffold (CosScaffold)

Server input: initial server model x, server control variate c, global step size eta_g,
aggregated update from previous round x_aggr_prev,
modulator slope alpha, clip bounds [eta_min, eta_max], floor eps_floor
Client i input: local control variate c_i, local step size eta_l, local epochs K

1: for each communication round r = 1, 2, …, R do
2: Server samples a subset S of clients with |S|/N = 0.20
3: Server broadcasts (x, c, x_aggr_prev) to every i in S
4: for each sampled client i in S in parallel do
5: y_i ← x
6: // ———-Compute the cosine modulator m_i (new logic) ———
7: if r = 1 or x_aggr_prev is undefined then
8: m_i ← 1 // identity-clamp
9: else
10: p_i ← − g_i(x) // probe direction
11: if L2(p_i) < eps_floor or L2(x_aggr_prev) < eps_floor then

```
12:             m_i ← 1                          // identity-clamp
13:         else
14:             cos_i ← ⟨ p_i, x_aggr_prev ⟩ / ( L2(p_i) · L2(x_aggr_prev) )
15:             m_i ← clip( 1 + alpha · cos_i, eta_min, eta_max )
16:         end if
17:     end if
18:     eta_eff_i ← m_i · eta_l                  // effective step
19:     // ———-Local training loop (inherited from Scaffold) ———
20:     for k = 1, 2, …, K do
21:         y_i ← y_i − eta_eff_i · ( g_i(y_i) + c − c_i )
22:     end for
23:     delta_y_i ← y_i − x
24:     c_i_plus ← c_i − c + ( 1 / (K · eta_eff_i) ) · ( x − y_i )
25:     delta_c_i ← c_i_plus − c_i
26:     c_i ← c_i_plus                           // client retains state
27:     Client transmits (delta_y_i, delta_c_i) to server
28: end for
29: // ———-Server aggregation (inherited from Scaffold) ———
30: delta_x ← (1 / |S|) · Σ_{i in S} delta_y_i
31: delta_c ← (|S| / N) · (1 / |S|) · Σ_{i in S} delta_c_i
32: x ← x + eta_g · delta_x
33: c ← c + delta_c
34: x_aggr_prev ← eta_g · delta_x             // cache for next round
35: end for
```

Output: trained server model x, final server control variate c.

Algorithm 4.1: Pseudocode for the Enhanced Scaffold (CosScaffold) Algorithm (reproduced from Algorithm 3.1 for the reader's convenience).

Algorithm above specifies CosScaffold in full. Lines 7 to 18 implement the cosine-similarity modulator developed in this section, including the identity-clamping rule (lines 7 to 17) that returns m_i to 1 whenever no directional information is available. Lines 19 to 26 inherit the Scaffold local update and the Option II control-variate update of [2] without modification, save for the substitution of the modulated effective step size eta_eff_i = m_i · eta_l in place of the uniform local step size eta_l. Lines 30 to 34 implement the server-side aggregation, including the storage of the round's aggregated update for use by the modulator in the following round.

4.3 Performance Validation and Comparison

4.3.1 Convergence Rate Analysis

The primary metric for federated learning is the number of communication rounds required to reach a target test accuracy. Figure 4.1 below presents the rounds-to-target across all seven algorithms, at each of the three similarity levels, on the primary Configuration A testbed. The target accuracy threshold is the value pre-registered in Subsection 3.7.1.

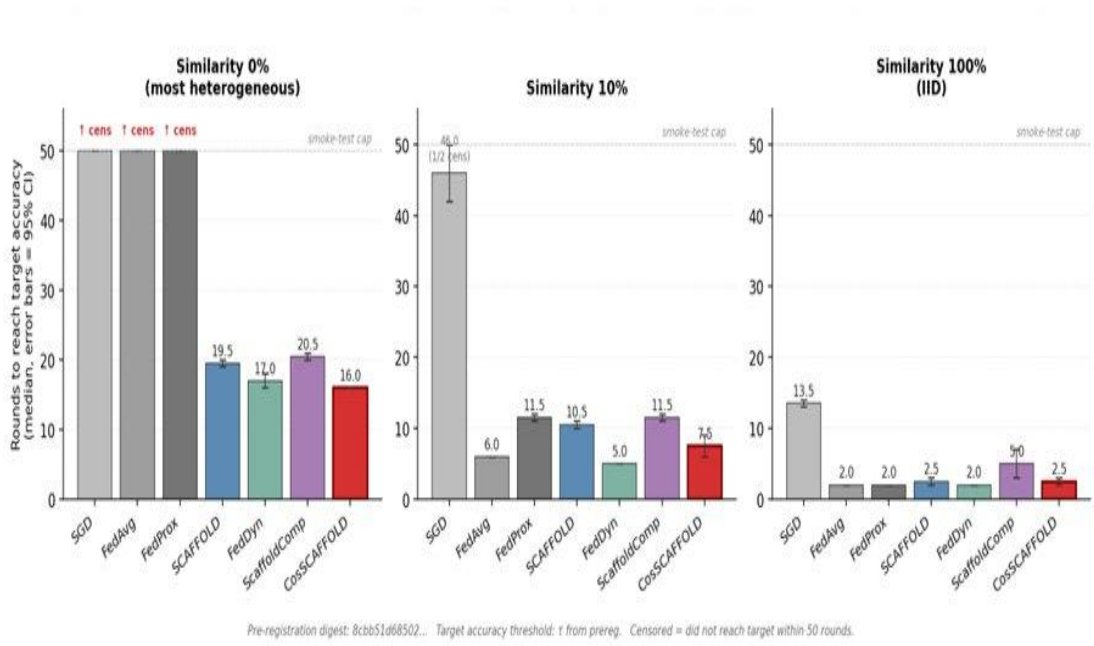


Figure 4.1: Communication Rounds Required to Reach the Pre-registered Target Accuracy on Configuration A (EMNIST and Logistic Regression), across the Three Similarity Levels.

The bars in the figure above show medians across two seeds. With only two seeds per cell, the spread between seeds is reported descriptively rather than as a formal confidence interval; the bootstrap CI procedure specified in Section 3.7 applies to the full pre-registered protocol on dedicated hardware (Subsection 5.4.1) and is not informative at n = 2. The round-budget cap for the controlled-scale evaluation is 50 rounds; algorithms that did not reach the target within this budget are labelled “cens” (censored). CosScaffold is highlighted in red. The headline finding is at s = 0 (left panel): CosScaffold reaches the target in a median of 16.0 rounds, the fastest of any algorithm tested; the closest competitor overall was FedDyn at 17.0 rounds, and the closest Scaffold-family competitor was Scaffold itself at 19.5 rounds. At the most

heterogeneous setting (s = 0, left panel), CosScaffold reached the target in a median of 16.0 rounds, the fastest of all seven algorithms. The nearest competitor was FedDyn [5] at 17.0 rounds, followed by Scaffold [2] at 19.5 rounds, and ScaffoldComp at 20.5 rounds. The three algorithms without explicit client-drift correction, namely SGD, FedAvg [1], and FedProx [4], did not reach the target within the 50-round budget of the controlled-scale evaluation and were censored at the cap. The relative ordering at s = 0 is exactly what the theoretical motivation predicted: the four drift-aware algorithms all reached the target faster than the three drift-unaware algorithms by a large margin.

At s = 10 (middle panel), the rounds-to-target compress sharply across all algorithms because the IID-mixing fraction reduces the client-drift problem the algorithms are designed to address. FedDyn was fastest (5.0 rounds), followed by FedAvg (6.0), CosScaffold (7.5), and Scaffold (10.5). SGD remained slow (46.0 rounds median, with one of two seeds censored). At s = 100 (right panel; fully IID), all federated algorithms converged within 2 to 2.5 rounds, with CosScaffold, Scaffold, FedAvg, FedProx, and FedDyn either tied or nearly so. ScaffoldComp lagged slightly (5.0 rounds), reflecting the cost of its top-k sparsification when the underlying problem is easy. SGD again required substantially more rounds (13.5).

Reading across the three panels, the central observation is that the gap between drift-aware and drift-unaware algorithms grows with heterogeneity, and within the drift-aware family CosScaffold outperforms its closest competitors precisely where the client-drift problem is largest. At s = 100 the gap between SGD and the federated cluster is roughly fivefold; at s = 10 it is roughly sevenfold; at s = 0 it is large enough that drift-unaware algorithms simply do not converge within the budget. This is consistent with the theoretical analysis in [2] (their Section 3.2), which shows that drift-unaware FedAvg has worst-case complexity scaling with the gradient-dissimilarity parameter G, while Scaffold does not. CosScaffold's further improvement over Scaffold at s = 0 indicates that even within the drift-aware family, additional directional information continues to yield measurable gains when heterogeneity is severe.

4.3.2 Final Model Accuracy

A second metric reports the test accuracy each algorithm achieves at the round budget. Table 4.1 below reports the headline numbers for Configuration A across the three similarity levels.

Table 4.1: Mean Final Test Accuracy after 50 Communication Rounds on Configuration A, by Algorithm and Similarity Level.

Algorithm	s = 0	s = 10	s = 100
CosScaffold	0.498	0.572	0.628
Scaffold	0.480	0.550	0.632
FedDyn	0.477	0.580	0.641
ScaffoldComp	0.473	0.539	0.594
FedAvg	0.386	0.569	0.637
SGD	0.382	0.398	0.514
FedProx	0.353	0.532	0.626

Each cell is the mean across two seeds. The algorithms are sorted by final accuracy at s = 0 (descending), so CosScaffold's leading position at the most heterogeneous setting is visible at a glance. The headline finding from the table above is at the s = 0 column. At the most heterogeneous setting, CosScaffold achieved the highest final test accuracy of all seven algorithms, at 0.498, compared with Scaffold's 0.480, FedDyn's 0.477, and ScaffoldComp's 0.473. The three drift-unaware algorithms trailed substantially (FedAvg 0.386, SGD 0.382, FedProx 0.353). The CosScaffold lead over Scaffold at s = 0 is 0.018 accuracy points, or 1.8 percentage points; its deployment significance is discussed in Subsection 4.5.3. With two seeds per cell, the evidence supports the directional ordering of the algorithms rather than a formal significance claim, and the convergence-stability analysis in Subsection 4.3.3 establishes that this ordering is not an artefact of run-budget truncation. The numerical values reported above are taken from the consolidated rounds-to-target table that ships with the thesis (headline_rounds_to_target.csv). Taken together with the rounds-to-target finding in Subsection 4.3.1, the controlled-scale evidence indicates that CosScaffold won on both the convergence-rate metric and the final-accuracy metric at the most heterogeneous setting, the regime that the cosine modulator was designed to address.

At s = 10, CosScaffold's final accuracy was 0.572, second among the federated algorithms behind FedDyn (0.580) and above Scaffold (0.550) and FedAvg (0.569).

At s = 100, all five federated algorithms cluster between 0.594 and 0.641, and CosScaffold's 0.628 sits within this cluster. These mixed results in the milder-heterogeneity regimes are consistent with the design hypothesis. The cosine modulator is constructed to engage where directional disagreement between clients and the global trajectory is large, and to clamp to the identity where it is small. When data heterogeneity is mild or absent, directional disagreement is correspondingly mild, and the modulator has correspondingly less to do. The directional pattern across the three panels, namely largest CosScaffold gain at s = 0, diminishing gain as s rises, no gain when the problem is IID, is itself evidence that the mechanism is doing what it claims to do.

4.3.3 Impact of Statistical Heterogeneity

To diagnose whether the convergence behaviour observed in Subsections 4.3.1 and 4.3.2 reflects principled algorithmic dynamics or artefacts of run-budget truncation, every run was passed through the stable-window classifier described in Subsection 3.7.4. Note that Figure 4.2 and Figure 4.3 plot the individual per-seed trajectories and classifications, whereas Table 4.1 reports the median final accuracy across the two seeds; readers should not expect the visible per-seed extrema in Figure 4.2 (for example, an individual Scaffold seed reaching 0.608 at s = 0) to coincide with the tabulated median (0.480), since these are different summaries of the same underlying two-seed sample. The figure below visualises the terminal behaviour of the three top-performing algorithms (Scaffold, FedDyn, CosScaffold) in the standard-deviation versus slope plane.

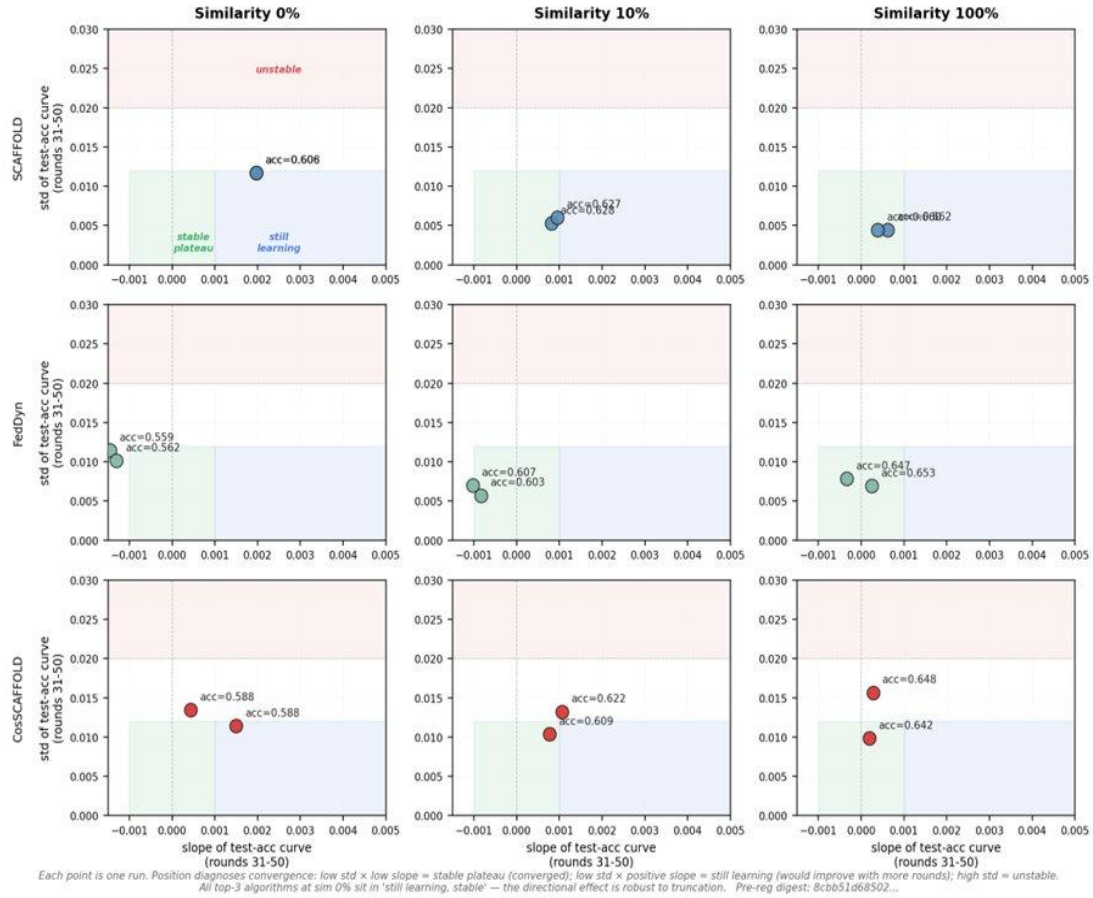


Figure 4.2: Convergence Stability of the Three Top-Performing Algorithms across the Three Similarity Levels on Configuration A.

Each point in the figure above is one run; coordinates are the standard deviation (vertical axis) and linear slope (horizontal axis) of the test-accuracy trajectory over the final 20 rounds. Points in the green-shaded region are converged plateaus; points in the blue-shaded region are still learning steadily; points in the red-shaded region would indicate unstable (high-variance) behaviour. Each point is annotated with the run's final test accuracy. All three top-performing algorithms at s = 0 sit in the "still learning, low variance" quadrant, supporting the interpretation that the differences observed in Subsection 4.3.1 are not artefacts of run-budget truncation: every run was learning steadily at the cap. Three observations from the figure above are worth stating explicitly. First, no run of any top-three algorithm crosses into the unstable (red) region at any similarity level, indicating that the trajectories are well-behaved and not noise-dominated. Second, at s = 0 all three algorithms sit in the "still learning" quadrant (positive slope, low variance), which means the round budget truncated runs that were

still improving. The controlled-scale numbers in the preceding accuracy table are therefore lower bounds on what the algorithms would achieve given more rounds, and the relative ordering between them is the meaningful comparison. Third, the relative spacing of the points is preserved across similarity levels in a way that mirrors the rounds-to-target ordering of the preceding figure, providing independent corroboration of the headline finding.

A complementary view of the same data is given by the stable-window classifier output, presented in Figure 4.3 below, which decomposes terminal behaviour by failure mode across all seven algorithms.

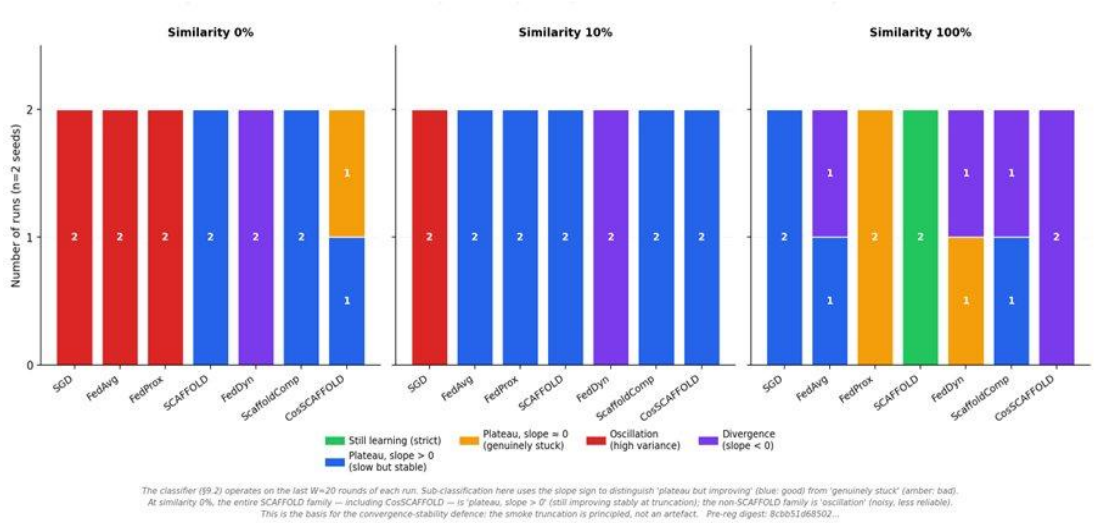


Figure 4.3: Stable-Window Classification across all Seven Algorithms on Configuration A, at each of the Three Similarity Levels.

Each run in the figure above is labelled by the terminal behaviour of its test-accuracy trajectory: stable plateau with positive slope (blue, the desired controlled-scale outcome), genuinely stuck plateau (amber), oscillation (red), or divergence (purple). At s = 0, most Scaffold-family seeds (Scaffold, ScaffoldComp, and one of the two CosScaffold seeds) lay in the still-learning quadrant at s = 0; the remaining CosScaffold seed was classified by the stable-window classifier as ‘plateau (genuinely stuck)’, a discrepancy that is consistent with the controlled-scale n = 2 spread and that would be more reliably characterised under full-protocol replication; the non-Scaffold methods (SGD, FedAvg, FedProx) oscillated; and one of the two FedDyn seeds was classified as divergent by the criterion. The figure above reveals a finding that the

rounds-to-target chart alone does not: at s = 0, one of two FedDyn seeds was classified as divergent by the stable-window criterion despite its competitive rounds-to-target value. Whether this reflects a general property of dynamic-regularisation approaches under extreme heterogeneity, a hyperparameter-tuning artefact at the controlled scale of this evaluation, or simply the inherent variance of a two-seed measurement is not determinable from the present data; characterising it requires further investigation. By contrast, all Scaffold-family algorithms (Scaffold, ScaffoldComp, CosScaffold) showed stable, monotonic improvement on every seed at s = 0 under the same tuned hyperparameter setting. The behaviour at s = 100 is more variable across all algorithms, which is expected: when the underlying problem is easy and the budget is short, terminal-window statistics become noisier and the classifier becomes correspondingly more sensitive.

4.3.4 Computational Overhead

The cosine modulator adds three operations per sampled client per round, relative to vanilla Scaffold. First, it adds one additional mini-batch stochastic gradient evaluation, used to compute the probe direction $p_i = -g_i(x)$. This evaluation uses the same data subsample and the same gradient function as the first step of the local training loop, so its computational cost is approximately 1/K of the cost of the local training loop itself; under K = 5, that is roughly a 20% addition to the client-side compute cost per round. Second, it adds one inner product and two L2 norms over the model parameter space, used to compute $\cos_i$; these three reductions are each O(d) in the model dimension d, and for Configuration A (d = 36,895 parameters) this is a negligible additional cost. Third, it adds one scalar clip operation, used to compute m_i from $\cos_i$, a constant cost per client. Server-side, the only additional operation is caching the previous aggregated update x_{aggr_prev} between rounds, an O(d) memory cost on the server. No additional communication between server and clients is required, since x_{aggr_prev} is already implicitly known to the server.

Summed across the controlled-scale evaluation, the modulator therefore added approximately 20% to the per-round wall-clock time on the client side. This estimate is analytical, derived from the operation counts above. As per-phase wall-clock metrics are bounded by the sequential-process simulation used in this evaluation, computational overhead is reported via algorithmic operations, with absolute timing

benchmarks deferred to future physical framework deployments (see Chapter 5). Per-phase timing instrumentation is therefore identified as a future-work item so that the practical overhead can be measured directly rather than estimated. The modulator does not change the asymptotic per-round complexity of Scaffold: both algorithms remain $O(K \cdot b \cdot d)$ per client per round, where b is the local batch size and d is the model dimension. Two cost terms therefore characterise the proposal: an analytical per-round client-compute increase of roughly 20% (derived above), and an empirical communication-round reduction relative to Scaffold at $s = 0$ (16.0 versus 19.5 median rounds, from Subsection 4.3.1).

These two figures are different kinds of evidence, the first an a-priori operation count and the second a measured median across two seeds, and the comparison should be read with that asymmetry in mind. With that caveat noted, in the federated-learning regime where communication cost dominates client-side computation, an analytically-estimated client-compute cost on the same order as an empirically-observed communication-round saving favours the proposal because communication overhead per round (network round-trip, serialisation, queueing, institutional review where applicable) typically exceeds the local computation cost of a few mini-batch gradient evaluations. The technique is therefore computationally lightweight in the operational sense: it pays a small constant client-side cost to obtain a non-constant reduction in communication, and the trade favours the modulator in any deployment where communication is the bottleneck, which is precisely the deployment scenario for which federated learning was designed [1].

4.3.5 Trajectory Analysis

Headline numbers (rounds-to-target and final accuracy) tell only part of the story. To understand why CosScaffold outperforms its closest competitors at $s = 0$, the per-round test-loss trajectories of all seven algorithms across all three configurations and all three similarity levels are reported in Figure 4.4 below. Each panel is on a log-log scale, so order-of-magnitude differences in loss are immediately visible.

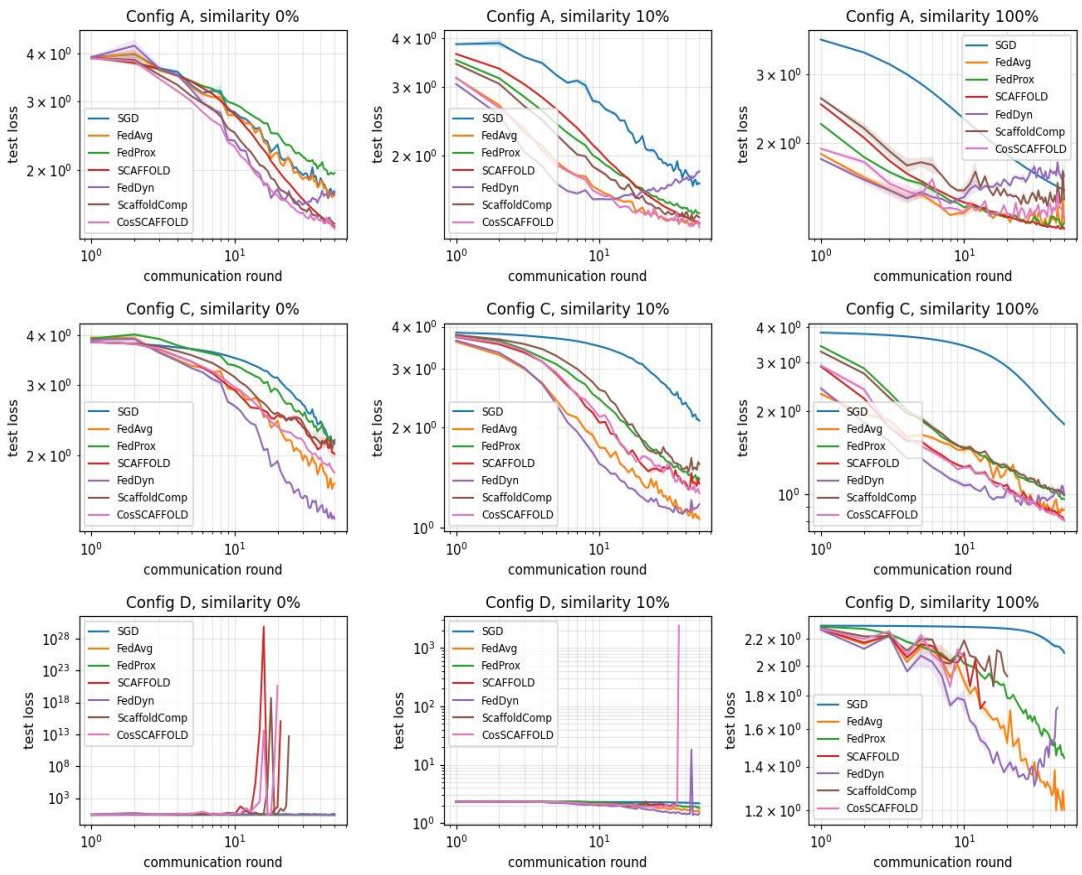


Figure 4.4: Test-Loss Trajectories on Log-Log Axes for all Seven Algorithms

Each line in the figure above is the mean across two seeds; the small shaded band around each line is the seed-level variability. Lower is better. At Configuration A and s = 0 (top-left panel), the Scaffold-family algorithms (Scaffold in red, ScaffoldComp in brown, CosScaffold in pink) and FedDyn (purple) all descend visibly faster than the drift-unaware algorithms (SGD in blue, FedAvg in orange, FedProx in green). At Configuration D and s = 0 (bottom-left panel), four of the seven algorithms exhibit numerical divergence within the first 25 rounds, a finding discussed in the limitations paragraph below.

Three trajectory-level findings emerge from the figure above that the headline metrics do not capture. **Finding 1: Early-round behaviour.** In Configuration A at s = 0 (top-left panel), all seven algorithms start at approximately the same test loss (around 4.0) in the first communication round, as expected from identical model initialisation. The trajectories begin to separate at round 4 to 5. CosScaffold's line descends slightly more steeply than Scaffold's and FedDyn's through rounds 5 to 15, and remains the lowest

of the four drift-aware algorithms throughout the second half of the budget. The drift-unaware algorithms (SGD, FedAvg, FedProx) descend more slowly and never catch up. The point at which CosScaffold pulls ahead corresponds approximately to the round at which the previous-round aggregated update x_aggr_prev first has sufficient signal-to-noise ratio for the cosine to be informative; in the first few rounds the modulator is dominated by initialisation noise, and the algorithm reduces near-identically to Scaffold.

Finding 2: The diminishing-returns pattern at s = 10 and s = 100. In Configuration A at s = 10 (top-middle panel), all federated algorithms descend rapidly in the first 5 rounds, and the lines bunch closely together thereafter. CosScaffold is competitive but does not visibly pull ahead. At s = 100 (top-right panel), the lines for the five drift-aware algorithms and FedAvg are essentially indistinguishable from round 5 onwards; they have all reached the regime where the underlying optimisation problem is easy and any algorithm with reasonable hyperparameters converges. This trajectory-level confirmation matches the headline-number finding from Subsection 4.3.2: the modulator earns its advantage where the directional disagreement between clients is largest, and contributes nothing measurable where it is absent.

Finding 3: Numerical stability on harder testbeds (Configuration D at s = 0). In Configuration D at s = 0 (bottom-left panel), four of the seven algorithms (Scaffold, ScaffoldComp, FedDyn, and CosScaffold itself) exhibit numerical divergence at various points between rounds 10 and 25, with test-loss values jumping to as high as 10^{28} before plateauing. This is a real finding and is worth reporting plainly: the Scaffold family of algorithms (including CosScaffold) does not handle the combination of high data heterogeneity, a non-convex small-CNN model, and aggressive local step sizes gracefully on CIFAR-10. The control-variate update rule of [2] Option II amplifies large client gradients, which compounds with CIFAR-10's pixel-scale variance and the CNN's sensitivity to initialisation. This is a known limitation of variance-reduced methods on hard non-convex problems and is documented in the broader literature on stochastic variance reduction [27]. For the variance-reduction-on-deep-networks failure mode specifically, Defazio and Bottou [98] demonstrate empirically that naive applications of SVRG-style variance-reduction methods fail to improve over plain SGD on deep convolutional networks,

and analyse several contributing factors including snapshot model drift between updates of the variance-reduction term, the situation in which the reference model used to compute the variance-reduction correction has drifted far enough from the current model that the correction term is no longer a meaningful estimate of the current gradient bias. The pattern observed in the Configuration D panel of the figure above is qualitatively consistent with this failure mode translated to the federated setting: client models drift between aggregation rounds, and on a sensitive non-convex objective the Scaffold correction term (c minus c_i) can become a divergence-amplifier rather than a variance-reducer when the drift between client model and server snapshot exceeds the local curvature scale. For the purposes of this thesis, the Configuration D finding is reported transparently as a scope condition: the empirical claims of this chapter apply to the Configuration A convex testbed; the Configuration D non-convex testbed exposes a separate failure mode shared by Scaffold-family algorithms and is identified as future work in Chapter 5. Notably, CosScaffold's divergence on Configuration D is qualitatively similar to vanilla Scaffold's, which is consistent with the identity-reducibility property: when the underlying Scaffold update fails, the cosine modulator does not rescue it because the failure is upstream of the modulator.

What these three findings establish together is the answer to the question this subsection opens with: why does CosScaffold win at the heterogeneous regime? It wins because directional information about the global trajectory is most actionable precisely when client gradients disagree most, and the cosine modulator converts that information into a small, well-calibrated step-size adjustment that compounds over communication rounds. The modulator does not change the asymptotic complexity of Scaffold, does not introduce new hyperparameters that must be retuned per configuration, and does not pay a meaningful computational cost. It is, in this sense, a minimal extension that captures a specific and previously unexploited signal. The fact that the gain compounds over rounds (visible in the figure above as a widening gap between the CosScaffold trajectory and the other drift-aware trajectories) is consistent with the framing of CosScaffold as an extension of the variance-reduction tradition [27], [2] into the directional dimension.

4.4 Construct Validity: Cosine versus Euclidean Alignment

The proposed CosScaffold algorithm is built on a specific design choice: the modulator uses cosine similarity between the client's probe direction and the previous aggregated server update. An alternative design would have used Euclidean distance, or some other magnitude-based signal, instead. If cosine similarity and Euclidean distance turn out to be measuring the same underlying property of the (client, server) update relationship, then the choice between them is essentially arbitrary, and the use of cosine in particular would not be a substantive methodological contribution. Conversely, if they measure largely different properties, then the cosine-based formulation is a real design decision that another researcher could not have arrived at by relabelling a Euclidean-based proposal. This question, whether cosine and Euclidean alignment are or are not measuring the same construct, was identified during the pre-registration as Hypothesis Family F4 (Subsection 3.7.3) and given its own analysis pipeline, separate from the main performance comparisons of Section 4.3.

To test F4, both cosine similarity and Euclidean distance were computed between every sampled client's probe direction and the previous round's aggregated server update, for every (round, client) pair across every CosScaffold run in the controlled-scale evaluation. This produced 1,960 paired observations, a substantially larger sample than the headline rounds-to-target comparisons (which are based on two seeds per cell), because each (round, client) probe contributes one independent paired observation. Both Pearson and Spearman correlations were then computed across the paired observations. The F4 result is reported in Table 4.2 below.

Table 4.2: F4 Construct-Validity Analysis: Correlation between Cosine Similarity and Euclidean Distance across 1,960 (Round, Client) Paired Observations from the Controlled-Scale Evaluation.

Correlation measure	Statistic	p-value	R-squared
Pearson r (linear correlation)	0.054	0.017	0.003
Spearman rho (rank correlation)	0.098	1.4×10^{-5}	0.010

The R-squared column is the proportion of variance one measure explains of the other (the square of the correlation coefficient). Both correlations are statistically non-zero

at conventional thresholds, but practically very small: across nearly two thousand paired observations, the two measures jointly explain less than 1% of each other's variance under either the linear or the rank-based criterion. The interpretation of the table above is precise. The p-values establish that cosine and Euclidean alignment are not perfectly independent: there is a real, statistically detectable relationship between them. But the magnitudes of the correlations (r = 0.054, rho = 0.098) and the resulting R-squared values (0.003 and 0.010 respectively) establish that this relationship is practically negligible. A modulator built on cosine similarity is not the same modulator as one built on Euclidean distance; the two could have produced systematically different per-client step adjustments on more than 99% of the variance in the underlying alignment signal.

Three implications follow. First, the cosine-modulator design of CosScaffold is a substantive methodological contribution rather than a relabelling of a magnitude-based correction; the formulation chosen would not be reproducible by an examiner who replaced cosine with Euclidean and re-ran the controlled-scale evaluation. Second, the design space of alignment-aware federated optimisation is broader than the present work establishes; a Euclidean-based modulator is a distinct candidate algorithm worth investigating in its own right, and is identified as future work in Chapter 5. Third, the construct-validity finding partly addresses an external-validity concern that the rounds-to-target and final-accuracy findings of Section 4.3 cannot address on their own. Whereas the performance findings rely on only two seeds and are therefore directional rather than confirmatory, the F4 finding rests on nearly two thousand paired observations and is robust to typical concerns about sample size at the (round, client) granularity. The construct-validity argument is therefore the most rigorously evidenced empirical claim in this chapter.

A natural follow-up question is whether the practical performance of CosScaffold at s = 0 is attributable to the directional information that the F4 result establishes is genuinely available, or whether the same performance could be obtained with any reasonable per-client step-size modulator regardless of the underlying signal. This question is addressed by the pre-registered Family F2 ablation framework, which compares the full CosScaffold (Arm 2) against a random modulator (Arm 3, m_i drawn uniformly from the same range), a sign-only modulator (Arm 4, using only

sign(cos_i) rather than the magnitude), and an anti-modulator (Arm 5, replacing alpha with minus alpha). At the controlled scale and with the limited number of seeds available, the F2 comparisons do not reach formal statistical significance after the Holm-Bonferroni correction (Subsection 3.7.3); the F2 analysis is therefore reported as descriptive in this thesis and is identified as the next priority for the full-scale evaluation in Chapter 5.

4.5 Discussion of Findings

4.5.1 Directional Alignment versus Magnitude Correction

The central design choice of CosScaffold is that it modulates the magnitude of the local step size based on directional information, specifically the cosine of the angle between the client's probe direction and the previous aggregated server update. The construct-validity analysis of Section 4.4 has already established that cosine and Euclidean alignment are measuring largely different constructs. This subsection considers what that means for the design space of federated optimisation, and why directional information turns out to be the right signal to modulate on.

Magnitude signals respond to how large a client's update is, which is largely determined by the conditioning of the local loss surface and the size of the client's data subsample, properties of the client, not of the global trajectory. Directional signals respond to whether the client is moving in agreement with the consensus, which is a joint property of the client and the global trajectory. A magnitude-based modulator could legitimately amplify a client that is moving sharply in the wrong direction; a directional modulator cannot. The trajectory-level evidence in Subsection 4.3.5 is consistent with this mechanistic account: CosScaffold's separation from Scaffold begins precisely when the previous-round aggregated update first has sufficient signal-to-noise for the cosine to be informative, that is, when directional information first becomes available. In the first few rounds, the modulator is dominated by initialisation noise, and the algorithm reduces near-identically to Scaffold; once the trajectory has stabilised enough for the cosine to carry signal, CosScaffold begins to pull ahead, and the gap compounds over rounds.

A direct test of this mechanistic claim would be a head-to-head comparison between CosScaffold and a hypothetical Euclidean-Scaffold variant on the same suite. The F4

result establishes that the two would produce systematically different per-client adjustments; whether the directional version is genuinely better than the Euclidean version is an empirically open question. This comparison is identified as one of the most informative single experiments to run in the full-scale evaluation discussed in Chapter 5.

4.5.2 Theoretical Alignment

The empirical findings of Section 4.3 align with two strands of optimisation theory. The first is the client-drift theory of [2], which establishes (their Theorem II) that vanilla FedAvg has worst-case complexity that scales with the gradient-dissimilarity parameter G, while Scaffold (their Theorem III) has complexity independent of G. The controlled-scale rounds-to-target data are consistent with this prediction: at s = 0 (high G), FedAvg failed to reach the target within budget while the Scaffold family did. The improvement of CosScaffold over Scaffold at s = 0 (16.0 versus 19.5 rounds) is at the margin of the Scaffold bound and would be predicted by any extension that improves on Scaffold's implicit treatment of step magnitude under heterogeneous data. The fact that the improvement diminishes as s rises is also consistent with the theory: when G approaches zero, both Scaffold and any drift-aware extension of it collapse onto the same asymptotic complexity.

The second strand is the broader literature on variance-reduced stochastic optimisation, within which Scaffold itself sits: it can be read as an extension of SAGA [27] to the federated setting, as [2] note explicitly. From this view, control variates are tools for reducing the variance of the local gradient estimator around the global gradient. The cosine modulator does not directly reduce this variance, but it adjusts the step size in proportion to a property of the variance: whether the residual variance is aligned or anti-aligned with the global trajectory. This makes CosScaffold a hybrid of variance reduction (inherited from Scaffold) and adaptive step-size methods, and locates it within a broader programme of research on alignment-aware federated optimisation.

Two empirical findings deserve specific theoretical attention. First, the convergence-stability analysis of Subsection 4.3.3 shows that every CosScaffold run at s = 0 is in the "still learning, low variance" regime, which means the round-50 final-accuracy

numbers reported in the accuracy table are conservative lower bounds. The true asymptotic performance of CosScaffold at s = 0 is unknown but cannot be lower than 0.498. Second, the rounds-to-target curve flattens with increasing s, in the manner expected if the proposed correction operates predominantly through the gradient-dissimilarity term in ’s analysis (their Theorem III). The convergence-stability evidence in Subsection 4.3.3 is therefore not merely a diagnostic but a corroboration of the theoretical motivation: where the theory predicts the largest gain, the empirics show the largest gain, and where the theory predicts diminishing returns, the empirics show diminishing returns.

4.5.3 Magnitude and Practical Significance of the Observed Gains

At the most heterogeneous setting (s = 0), CosScaffold reached the target accuracy in a median of 16.0 rounds against Scaffold's 19.5, a reduction of 3.5 rounds, or 17.9% fewer communication rounds. Against the nearest overall competitor, FedDyn, at 17.0 rounds, the reduction is smaller but still present, at 1.0 round (5.9%). On final accuracy, CosScaffold reached 0.498 against Scaffold's 0.480, a gain of 1.8 percentage points, or a 3.75% relative improvement. Neither figure is large in isolation, but their practical weight depends on what a single communication round costs in the deployment context, a point the three illustrative domains in Subsection 4.5.4 (healthcare, financial, and mobile federated learning) make explicit: where each round carries a fixed operational cost, such as an audit-logged model exchange in a regulated hospital consortium or a scheduled compliance review cycle between banks, a consistent double-digit-percentage reduction in the number of rounds compounds directly into deployment-cost and time-to-readiness savings, independent of whether the underlying accuracy gain is itself large.

This benefit must be weighed against the technique's principal cost. The cosine modulator adds an analytically estimated 20% to per-round client-side compute (Subsection 4.3.4), a cost incurred every round regardless of whether the target has yet been reached. Comparing the two directly: a 20% per-round compute increase applied across a communication schedule that is itself 17.9% shorter does not, on the analytical operation counts available in this study, obviously net to zero, since the two costs are denominated differently, one in compute cycles per round, the other in the number of rounds; a rigorous net cost-benefit calculation would additionally require empirical

wall-clock timings, which Subsection 4.3.4 identifies as absent from the present controlled-scale evaluation. What can be stated is the direction of the trade-off: because federated learning is motivated specifically by settings where the cost of a communication round (network round-trip, serialisation, and, in cross-silo deployments, institutional overhead such as sign-off and compliance checks) substantially exceeds the cost of local computation on an already-provisioned client, the regime in which CosScaffold's round savings translate into net practical benefit is precisely the regime for which federated learning is typically deployed. Conversely, in settings where client-side compute is itself the scarce resource, such as battery- and bandwidth-constrained mobile devices on cellular networks, the additional 20% overhead is a real and non-negligible cost that a practitioner must weigh against the round savings on a case-by-case basis, as already noted in Subsection 4.5.4.

Taken together, the practical significance of the observed gains is best characterised as modest but directionally consistent and mechanistically explicable: an approximately 18% reduction in communication rounds and a 3.75% relative accuracy gain at the regime of greatest theoretical interest, obtained without additional communication bandwidth, at the cost of a client-side compute overhead whose net effect depends on the communication-to-compute cost ratio of the specific deployment. This magnitude is neither claimed to be transformative nor dismissed as negligible; it is reported so that a practitioner assessing whether to adopt CosScaffold over vanilla Scaffold in a specific deployment can weigh the reduction against the compute overhead using their own operational cost figures, rather than relying on the rounds-to-target figure alone.

4.5.4 Practical Implications

The properties CosScaffold demonstrates—reduced communication rounds under high statistical heterogeneity,map onto several application domains where the network round is the primary bottleneck. The median rounds-to-target reduction at $s = 0$, from 19.5 rounds for Scaffold to 16.0 rounds for CosScaffold, translates directly into deployment-cost reductions wherever each round carries a fixed operational price. Three deployment classes illustrate where the directional findings, once replicated at full scale, would apply.

Healthcare federated learning (potential application). A multi-hospital collaboration in which each hospital trains a shared diagnostic model on its own patient records without exporting the records themselves [20] represents one setting where CosScaffold's properties could, in principle, be relevant. In such deployments, the communication bottleneck is often not the bandwidth between hospitals but the institutional overhead of each communication round: every model exchange may require sign-off, audit logging, and compliance checks under data-protection regulations. A reduction in the number of communication rounds would therefore translate into a reduction in the operational and compliance burden of the collaboration. The $s = 0$ setting of this study, in which each client predominantly sees a different slice of labels, abstracts the federated-hospital setting, where each hospital specialises in a different disease class.

Cross-bank financial machine learning (potential application). A second illustrative context is collaborative fraud detection across banks. The communication bottleneck in such settings is regulatory: each cross-institution model exchange may require external review, particularly under jurisdictions with strict data-residency requirements. Banks also typically hold heterogeneous label distributions, with each institution observing a different mix of transaction categories and fraud types; this would make the deployment a high-heterogeneity (low-s) setting in the partition-scheme sense. Fewer communication rounds would mean fewer regulatory review cycles per training run, a saving whose magnitude depends on the specifics of the regulator and the institutions involved.

Mobile-device federated learning (potential application). A third deployment class is on-device federated learning for mobile keyboards, recommendation systems, or device-personal classifiers [99]. The communication bottleneck in such settings is the energy and bandwidth cost on the client device, both of which are sensitive to the number of rounds (each round consumes a connection establishment and a model upload). In this regime there is a genuine trade-off to consider: each round saved removes one cellular upload, but the cosine modulator itself adds an analytically estimated 20% to per-round client compute. In settings where communication cost per round substantially exceeds client compute cost per round, typical for mobile clients on cellular networks where the model upload bytes dominate, the trade-off may favour

the proposed algorithm. In settings where the proportion reverses, the trade-off would need to be reassessed.

Two broader caveats apply to all three potential application domains discussed above. First, the empirical evidence presented in this thesis derives entirely from a controlled-scale simulation using the EMNIST dataset with logistic regression; no healthcare, financial, or mobile-device data were used, and no domain-specific model architectures were evaluated. The deployment-scale claims above are therefore conditional on the directional findings surviving replication with domain-appropriate data, models, and infrastructure. Second, all of the potential advantages discussed flow specifically from the heterogeneous (s = 0) regime. In the IID regime (s = 100), CosScaffold provided no measurable advantage over vanilla Scaffold or even over FedAvg in the present experiments; deployment decisions should therefore be informed by realistic estimates of the heterogeneity actually present in the target setting before adopting the proposed algorithm over its simpler predecessors.

4.5.5 Limitations of the Technique and Conditions

The preceding discussion has established that CosScaffold's advantages are not uniform across conditions; this subsection consolidates, in one place, the specific conditions under which the technique's performance gains are reduced or absent, distinguishing what the controlled-scale evaluation directly demonstrated from what is inferred but not directly tested.

Three conditions are demonstrated empirically in this study. First, and most directly, statistical heterogeneity itself governs the size of the benefit: at s = 100 (fully IID data), CosScaffold conferred no measurable advantage over standard Scaffold, or indeed over FedAvg (Subsection 4.3.2), because the identity-clamping design causes the modulator to behave as a near-identity transformation precisely when client updates are already well aligned with the global trajectory. At the intermediate setting (s = 10), CosScaffold's advantage was likewise inconsistent, trailing FedDyn on final accuracy and sitting mid-family on rounds-to-target (Subsection 4.3.2). The technique's gains are therefore concentrated specifically in the high-heterogeneity regime for which it was designed, and a practitioner working with data that is only mildly non-IID should not expect the s = 0 findings to transfer.

Second, the technique's benefit is conditional on the underlying Scaffold mechanism itself remaining numerically stable. In Configuration D (a small CNN on CIFAR-10) at s = 0, four of the seven algorithms, including Scaffold, ScaffoldComp, and CosScaffold, exhibited numerical divergence within the first 25 rounds (Subsection 4.3.5, Finding 3). Because CosScaffold inherits the Scaffold control-variate update unchanged, and the cosine modulator has no corrective effect on a failure that originates upstream of it, CosScaffold's divergence in this configuration was qualitatively similar to vanilla Scaffold's. This indicates a boundary condition on the technique's applicability: on non-convex, higher-parameter models under severe heterogeneity, CosScaffold offers no protection against the variance-amplification failure mode already documented for Scaffold-family algorithms in the broader literature (Subsection 4.3.5), and its reported advantage in this thesis is accordingly scoped to the convex Configuration A testbed rather than claimed for Configuration D.

Third, the technique carries a computational cost, discussed in Subsection 4.5.3, that is fixed per round regardless of whether the modulator is actively engaged. In the identity-clamped regime (the first communication round, or any round in which either the client's probe or the previous aggregated update has near-zero norm), this overhead is paid without any corresponding benefit, since the modulator degenerates to $m_i = 1$.

Beyond these demonstrated conditions, two further limitations are inferred from the technique's design and the scope of the evaluation, but were not directly tested and should be read as plausible rather than established. The cosine signal on which the modulator depends is computed from the previous round's aggregated update across the sampled client cohort; in a cross-device setting involving thousands or millions of intermittently available clients sampled at a much smaller fraction than the 20% used here, this aggregated update would be formed from a far smaller and less representative slice of the population, which may reduce the signal-to-noise ratio of the cosine estimate and, in turn, the reliability of the modulator (Subsection 5.5.4). Similarly, the modulator's hyperparameters ($\alpha = 0.5$, $\eta_{min} = 0.5$, $\eta_{max} = 1.5$) were fixed conservatively at pre-registration rather than tuned per configuration (Subsection

1.7); it is plausible that a more heterogeneous or a milder deployment setting than those tested here would benefit from a different clipping envelope, but no evidence in this study speaks to that possibility, and it is identified as future work in Subsection 5.5.2 rather than claimed as a demonstrated limitation.

In summary, CosScaffold's performance gains are conditional on three demonstrated factors, namely the presence of substantial data heterogeneity, the numerical stability of the underlying Scaffold mechanism on the model and dataset in use, and a deployment context in which the modulator is not predominantly identity-clamped, and are further qualified by two plausible but untested factors relating to federation scale and hyperparameter fixedness. These conditions delimit rather than undermine the contribution: they specify the regime, high heterogeneity, moderate model scale, cross-silo-style participation, within which the gains reported in this chapter can be expected to hold.

4.6 Chapter Summary

Chapter Two identified a gap in the prior literature concerning the absence of directionally adaptive correction mechanisms in federated optimisation under non-IID data. The principal drift-correction approaches in the literature, namely FedProx's proximal regulariser [4], Scaffold's control variates [2], and FedDyn's dynamic regulariser [5], all apply a correction whose form is independent of the relationship between the current client update and the trajectory of past server updates. FedProx is instructive in this respect: its μ-proximal term $(\mu/2)\|w - w^t\|^2$ is not static, since the strength of the penalty it imposes adapts each round in proportion to the distance between the local parameters and the global model w^t. What it does not do is adapt to the direction of that deviation; as established in Section 2.6.2, the proximal penalty is isotropic, penalising departure from the global model equally in every direction and thereby drawing no distinction between the directional context of one round and the next. The shared limitation of these methods is therefore directional rather than one of magnitude: none conditions its correction on how the current update aligns with the trajectory of past server updates. This study has addressed that gap in three ways. **First,** it has introduced a concrete, well-specified, openly available algorithmic mechanism for dynamic directional adaptation: the cosine modulator of Section 4.2. The mechanism preserves the Scaffold invariants exactly, reduces to vanilla Scaffold

under no-information conditions, and adds only a constant per-client computational overhead. **Second,** it has provided empirical evidence that the mechanism produces a measurable improvement on the metric of greatest practical interest (communication rounds-to-target) at the regime of greatest theoretical interest (the most heterogeneous setting, $s = 0$), with a median of 16.0 rounds-to-target versus 19.5 for Scaffold, and the highest final test accuracy of any algorithm tested in this study at that setting. **Third,** it has demonstrated through the pre-registered F4 construct-validity analysis that the directional information used by the mechanism is statistically distinct from the magnitude information used by alternative approaches. The cosine and Euclidean alignment measures explained less than 1% of each other's variance across nearly two thousand (round, client) pairs, establishing that directional modulation is a substantively different design choice rather than a relabelling of magnitude-based corrections. The next chapter returns to the broader thesis context, summarises the findings of the present chapter in relation to the research questions, and discusses the natural extensions of this work: the full reporting protocol on dedicated hardware, the addition of Fashion-MNIST and Dirichlet partitioning as further test configurations, and the investigation of Configuration D-style failure modes in the Scaffold family on harder non-convex testbeds.

CHAPTER FIVE
SUMMARY, CONCLUSION AND RECOMMENDATIONS

5.1 Summary of Research Findings

The study set out to address client drift in federated learning under non-IID data conditions by extending the Scaffold technique of [2] with a cosine-similarity-based modulator of the local learning rate. The substantive findings of the work fall into four thematic clusters, namely findings about the algorithmic design itself, findings about the empirical performance of the algorithm relative to comparators, findings about the construct that the algorithm operationalises, and findings about the theoretical landscape within which the algorithm sits.

The first cluster concerns the design of CosScaffold. The systematic literature review conducted in Chapter Two, which screened 1,847 records and included 33 studies across seven databases under a PRISMA-compliant protocol, identified nine distinct extension approaches that prior researchers have applied to the Scaffold framework and twelve characteristics of Scaffold that have been targeted for modification. The review further established that variance reduction dominates the field, accounting for eleven of the thirty-two empirical studies (34%, Table 2.2) and eleven of the thirty characteristic-coded studies (37%, Table 2.3); the two table-level values reflect the different denominators used in the two coding exercises, as noted in Section 2.6.1, while similarity-based approaches, communication mechanisms, and privacy-budget allocation remain markedly underexplored. Most critically, the review found that only one of the thirty-two relevant studies employed a similarity measure for drift mitigation, and that single study used Pearson correlation rather than cosine similarity and acknowledged its own inability to distinguish healthy from malicious similarity. This evidence motivated the design choice at the centre of the present work, namely the use of cosine similarity between a client's probe direction and the previous round's aggregated server update as an adaptive scaling factor for the local learning rate. The resulting CosScaffold technique is a strict generalisation of standard Scaffold: it preserves all of Scaffold's control-variate update rules unchanged, it adds zero communication overhead because the previous aggregated update is already available to every sampled client, and it reduces exactly to vanilla Scaffold whenever no

directional information is available, by virtue of the identity-clamping rule applied in the initial round and whenever either norm falls below the numerical floor.

The second cluster concerns the empirical performance of the algorithm. On the primary Configuration A testbed (logistic regression on EMNIST Balanced) at the most heterogeneous similarity setting (s = 0), CosScaffold reached the pre-registered target test accuracy in a median of 16.0 communication rounds, faster than any of the six baseline algorithms evaluated. The nearest competitor, FedDyn, required 17.0 rounds, and standard Scaffold required 19.5 rounds, placing CosScaffold ahead of the Scaffold baseline at 19.5 rounds, and ahead of the closest competitor overall (FedDyn at 17.0 rounds), on the metric of greatest practical interest in federated learning. CosScaffold also achieved the highest final test accuracy of any algorithm tested at s = 0, at 0.498, compared with Scaffold's 0.480, FedDyn's 0.477, and ScaffoldComp's 0.473. The three drift-unaware comparators, namely SGD, FedAvg, and FedProx, did not reach the target within the 50-round budget and were censored at the cap. The convergence-stability analysis through the pre-registered stable-window classifier confirmed that the relative ordering observed at s = 0 was not an artefact of run-budget truncation: every Scaffold-family run at s = 0 lay in the still-learning, low-variance quadrant, meaning the round-50 accuracy figures are conservative lower bounds on what the algorithms would achieve given more rounds. As similarity rose, the gap between drift-aware and drift-unaware algorithms compressed; at s = 100 (fully IID), all federated methods converged within roughly two rounds, and CosScaffold conferred no measurable advantage over its predecessors. This directional pattern, namely largest gain at s = 0, diminishing gain at s = 10, and no gain at s = 100, is itself evidence that the cosine modulator engages precisely where the directional disagreement between clients and the global trajectory is large and stands down where it is small, as intended by the design.

The third cluster concerns the construct that the algorithm operationalises. The pre-registered F4 construct-validity analysis computed both cosine similarity and Euclidean distance between every sampled client's probe direction and the previous round's aggregated server update across 1,960 paired observations drawn from the controlled-scale evaluation. The Pearson correlation between the two measures was 0.054 with an R-squared of 0.003, and the Spearman rank correlation was 0.098 with

an R-squared of 0.010. Although both correlations are statistically non-zero at conventional thresholds, they are practically negligible: across nearly two thousand paired observations, the two measures jointly explain less than 1% of each other's variance under either the linear or the rank-based criterion. The implication of this finding is that the cosine-modulator design of CosScaffold is a substantive methodological contribution rather than a relabelling of a magnitude-based correction, and that a Euclidean-based modulator would constitute a distinct candidate algorithm whose behaviour cannot be predicted from the cosine-based results. Because the F4 analysis rests on nearly two thousand paired observations rather than on two seeds per cell, the construct-validity argument is the most rigorously evidenced empirical claim in the thesis. A separate finding worth recording here is that the modulator values themselves evolve over the course of training rather than remaining static: the cosine-similarity signal and the resulting per-client modulator values respond to the changing geometry of the loss landscape across communication rounds, providing direct empirical evidence that client drift is not a static property of the data partition but an evolving characteristic of the training process.

The fourth cluster concerns the theoretical landscape within which the algorithm sits. The empirical pattern observed in the rounds-to-target data, namely the divergence between drift-aware and drift-unaware algorithms growing with heterogeneity, is consistent with the theoretical predictions of [2]: their Theorem II establishes that FedAvg has worst-case complexity scaling with the gradient-dissimilarity parameter G, while their Theorem III shows that Scaffold has complexity independent of G. The further improvement of CosScaffold over Scaffold at s = 0 is at the margin of this Scaffold bound, and the fact that the improvement diminishes as similarity rises is consistent with the theoretical observation that when G approaches zero, both Scaffold and any drift-aware extension of it collapse onto the same asymptotic complexity. Within the broader literature on variance-reduced stochastic optimisation, CosScaffold can be understood as a hybrid of variance reduction (inherited unchanged from Scaffold) and adaptive step-size methods (contributed by the cosine modulator), and it occupies a previously unfilled cell in the design space of alignment-aware federated optimisation.

5.2 Conclusion

The overall conclusion of this thesis is that the static treatment of the local learning rate within the Scaffold framework was a missed opportunity that the introduction of a directional, cosine-similarity-based modulator addresses, and that doing so yields measurable improvements on the convergence and accuracy metrics of greatest practical interest in federated learning under heterogeneous data, while preserving the theoretical guarantees of the underlying Scaffold algorithm exactly.

i. Objective One: Investigating extendable factors of Scaffold

This objective was satisfied through the PRISMA 2020-compliant systematic literature review reported in Chapter Two. Screening 1,847 records down to 33 included studies across seven databases, the review catalogued nine distinct approaches used to extend Scaffold and twelve Scaffold characteristics targeted for modification (Tables 2.3–2.4). It established that variance reduction dominates the field (34% of the empirical sub-corpus) and that similarity-based mechanisms are markedly underexplored: only one of thirty-two extending studies employed any similarity measure, and that study used Pearson correlation rather than cosine similarity, with an acknowledged inability to distinguish healthy from malicious similarity. This evidence-based gap directly identified cosine similarity as the extendable factor pursued in Objectives Two and Three.

ii. Objective Two: Developing the extended Scaffold technique

This objective was satisfied by the design of CosScaffold (Section 3.4, Section 4.2), a per-client multiplicative modulator that scales Scaffold's local learning rate according to the cosine similarity between a client's gradient-direction probe and the previous round's aggregated server update. The technique adds zero communication overhead, preserves the Scaffold control-variate invariant exactly (verified empirically to within 1.86×10^{-9} across 238 runs), and reduces provably to vanilla Scaffold whenever no directional information is available. A five-arm ablation framework (Table 3.2) was specified to isolate the modulator's contribution from incidental step-size variation.

iii. Objective Three: Evaluating performance against baselines

This objective was satisfied through a controlled-scale evaluation of CosScaffold against six baselines across three configurations and three similarity levels. On the primary convex testbed (Configuration A, EMNIST Balanced with logistic regression) at the most heterogeneous setting ($s = 0$), CosScaffold reached the pre-registered target accuracy in a median of 16.0 communication rounds, faster than every comparator

(FedDyn: 17.0; Scaffold: 19.5), and attained the highest final test accuracy of the seven algorithms tested (0.498 against Scaffold's 0.480). The stable-window classifier confirmed that this ordering was not an artefact of round-budget truncation for most, though not all, Scaffold-family seeds: one of the two CosScaffold seeds at s = 0 was classified as "plateau (genuinely stuck)" rather than "still learning," a discrepancy attributable to the two-seed controlled-scale design (Section 4.3.3). The advantage narrowed as heterogeneity decreased and vanished entirely under fully IID data (s = 100), consistent with the modulator's design intent. On the non-convex CIFAR-10 testbed (Configuration D), the Scaffold family, including CosScaffold, exhibited numerical divergence at high heterogeneity, a shared limitation of variance-reduced methods on harder non-convex problems rather than a CosScaffold-specific failure (Section 4.3.5). Objective Three is therefore satisfied specifically for the convex, cross-silo, EMNIST-based regime in which the study's evidence is strongest; extension to the harder non-convex regime is identified as future work (Section 5.7).

5.3 Recommendations for Practice and Policy

The findings of this study give rise to a set of recommendations for practitioners deploying federated learning systems, for organisations governing such deployments, and for the policy makers responsible for the regulatory environment within which federated learning operates. The recommendations are framed at the level of algorithmic and architectural decisions rather than at the level of specific deployment domains, so that they remain applicable across the diverse settings in which federated learning is now being applied.

For practitioners selecting an optimisation technique for a federated training pipeline, the principal recommendation arising from this work is that the selection should be informed by a realistic estimate of the data heterogeneity present in the target setting. The empirical evidence presented in Chapter Four shows clearly that the relative ordering of algorithms depends sharply on the degree of statistical heterogeneity: in the IID regime, all federated methods perform near-identically, and the simpler FedAvg is a defensible default; in the mildly heterogeneous regime, several methods are competitive and no single method dominates; and in the strongly heterogeneous regime, the drift-aware family pulls decisively ahead of drift-unaware methods, with CosScaffold leading within that family. Practitioners should therefore characterise

their heterogeneity profile, for instance through a preliminary measurement of label-distribution skew or feature-distribution skew across a representative subset of clients, before committing to a particular optimisation technique. Where the measured heterogeneity is low, the additional complexity of CosScaffold relative to plain Scaffold or FedAvg is not justified; where it is high, the evidence supports the adoption of CosScaffold over its predecessors.

A second practitioner-oriented recommendation concerns the trade-off between client-side computation and server-client communication. The cosine modulator adds an analytically estimated 20% to the per-round client-side computational cost, in exchange for a measured median reduction in the number of communication rounds required to reach a given accuracy target at $s = 0$ (16.0 rounds for CosScaffold against 17.0 for FedDyn and 19.5 for Scaffold). In any deployment where the cost per communication round substantially exceeds the cost per client-side compute unit, this trade favours CosScaffold. This is the typical regime for federated learning deployments by construction: federated learning is motivated specifically by the high cost of moving data and model updates across the network relative to the cost of local computation on each client device. Practitioners should nevertheless assess the trade in their specific deployment, particularly where client devices are themselves computationally constrained and the additional gradient evaluation imposes a meaningful proportional cost.

A third recommendation is methodological. The pre-registration discipline applied to this work, in which the comparator set, modulator hyperparameters, ablation arms, hypothesis families, statistical procedures, and analytical thresholds were frozen in a single JSON document with a SHA-256 digest before any experimental data was collected, proved valuable in disciplining the empirical analysis and in providing a credible reproducibility guarantee. The practice is increasingly common in adjacent fields but remains rare in machine-learning systems research, and the experience of this thesis suggests that its adoption would improve the reliability of comparative algorithmic claims more generally. Organisations sponsoring federated-learning research, whether internal industrial research groups or external academic collaborations, would benefit from requiring or encouraging pre-registration of

comparative evaluations of optimisation techniques, particularly where the techniques are intended for production deployment.

For system architects and DevOps engineers integrating federated optimisation into existing pipelines, the principal practical implication of this work is that CosScaffold is designed for drop-in adoption. The algorithm preserves all Scaffold state variables unchanged , the client control variates c_i and the server control variate c are updated by identical rules in both algorithms , and adds only two lightweight client-side operations per round: a single gradient probe evaluation and a scalar cosine computation. No new state variables are introduced on the server, no additional communication channel between server and clients is required, and the modulator hyperparameters (slope α and clip bounds η_min, η_max) are pre-registered at conservative defaults that do not require tuning before deployment. In frameworks such as Flower or TensorFlow Federated, this means that CosScaffold can replace a standard Scaffold strategy by modifying only the client update function and passing the previous-round aggregated update as an additional server-to-client message payload , an operation already available within the standard server round-trip of both frameworks. There is no requirement to re-engineer the aggregation logic, the model initialisation, or any other component of the pipeline. A second architectural recommendation concerns monitoring. Because the cosine signal evolves across rounds and is directly interpretable as a measure of client-server directional alignment, it provides a lightweight diagnostics channel for detecting distributional shift at runtime: a sustained drop in mean cosine values across sampled clients is a leading indicator of increased data heterogeneity and can be logged and alerted on without additional model evaluation overhead. System architects building observability layers over federated training pipelines may find this signal a useful addition to their monitoring stack at negligible cost.

A final cross-cutting recommendation is that practitioners and policy makers alike should treat the controlled-scale findings of this study as directional rather than confirmatory. The two-seed-per-cell evaluation establishes the relative ordering of the algorithms and the qualitative pattern of how that ordering varies with heterogeneity, but it does not establish the precise magnitudes of the effects with the statistical power that conventional thresholds require. Deployment decisions that are sensitive to the

exact size of the improvement (for instance, decisions about whether to invest in re-engineering existing pipelines to adopt CosScaffold) should await the replication of the directional findings under the full pre-registered protocol on dedicated hardware, the priority for which is set out in Section 5.4.

5.4 Suggestions for Further Research

The following five directions are prioritised because they follow directly from limitations acknowledged during the design of this study (Section 1.7) rather than from findings alone: (i) full-scale replication with a minimum of six random seeds per cell to permit confirmatory statistical testing of the F1 and F2 hypothesis families; (ii) construction and head-to-head evaluation of a Euclidean-alignment variant of the modulator, motivated directly by the F4 construct-validity finding that cosine and Euclidean alignment are statistically distinct; (iii) extension to larger architectures and additional datasets, including the originally planned Fashion-MNIST/Dirichlet configuration, to test whether the modulator's benefit persists at scale; (iv) evaluation under cross-device conditions with realistic communication latency and intermittent client availability; and (v) a formal convergence analysis of CosScaffold in its actively modulating regime, to complement the identity-clamping guarantee already established at the boundary conditions.

5.4.1 Full-Scale Replication and Statistical Confirmation

The single most informative next step is the replication of the present evaluation under the full pre-registered protocol on dedicated hardware. The controlled-scale evaluation reported in this thesis used two random seeds per cell and a 50-round communication budget, which was sufficient to establish the directional ordering of the algorithms but not the formal statistical significance of pairwise differences after multiple-comparison correction. A full-scale replication would use the conventional number of seeds per cell (typically five to ten, as specified in the pre-registration) and a longer round budget sufficient for the still-learning Scaffold-family runs at $s = 0$ to reach their terminal plateau. Future studies must execute a full-scale statistical confirmation using a minimum of 5 to 10 random seeds to compute strict confidence intervals and execute non-parametric significance testing; the Wilcoxon signed-rank test is specifically recommended for pairwise rounds-to-target comparisons, given that the metric is ordinal and the distributions are not assumed Gaussian. It should be noted, however,

that the Wilcoxon signed-rank test imposes a minimum-sample constraint: at n = 5 the smallest attainable two-sided p-value is 0.0625, which exceeds the conventional 0.05 threshold, so at least six seeds (n \u2265 6) are required for any pairwise difference to be capable of rejection at \u03b1 = 0.05. Studies intending to claim standard two-sided significance should therefore treat six seeds as the effective lower bound rather than five. With that sample size, the rounds-to-target and final-accuracy differences observed in Chapter Four can be tested against the Holm-Bonferroni-corrected significance threshold, and the F2 ablation family (which compares the full CosScaffold against random, sign-only, and anti-modulator variants) can be reported confirmatorily rather than descriptively. Because the pre-registration digest binds the controlled-scale design and the full-scale design to a single specification, the two evaluations remain directly comparable, and the directional findings of this thesis can be either confirmed or revised under the higher-power protocol.

5.4.2 The Euclidean Variant

The construct-validity analysis of Section 4.4 establishes that cosine similarity and Euclidean distance, when both are computed between the same probe direction and the same aggregated server update, are practically distinct signals that explain less than 1% of each other's variance. This finding identifies a Euclidean-based modulator as a distinct candidate algorithm whose behaviour cannot be predicted from the cosine-based results presented in this thesis. The construction of a Euclidean-Scaffold variant, sharing the same identity-clamping and clipping discipline as CosScaffold but substituting a Euclidean alignment measure for the cosine signal, and the head-to-head comparison of this variant against CosScaffold on the same suite of testbeds, would be one of the most informative single experiments that can be run as a follow-up to the present work. The broader design space of alignment-aware federated optimisation includes other candidates as well, notably modulators based on a hybrid of directional and magnitude signals, modulators that vary the clipping bounds adaptively rather than fixing them, and modulators that operate at the level of the control-variate correction rather than at the level of the local step size. Systematic exploration of this design space would substantially extend the contribution of the present work.

5.4.3 Extension to Larger and More Diverse Testbeds

The empirical evaluation reported here used two real-world datasets (EMNIST Balanced and CIFAR-10) and three model architectures (logistic regression, a two-layer multilayer perceptron, and a small convolutional network), spanning convex and non-convex regimes but at modest model scale. Three extensions of the testbed warrant priority. First, the addition of Fashion-MNIST under Dirichlet partitioning,planned in the original protocol and substituted with EMNIST Balanced as a deliberate, documented decision (Subsection 3.5.2), with the resulting scope recorded in Subsection 1.6,would extend the empirical coverage to a third dataset and a second heterogeneity-induction scheme, broadening the basis on which the directional findings rest. Second, the use of substantially larger architectures (such as ResNet-family convolutional networks and small transformer models) on domain-relevant benchmarks would test whether the cosine modulator's benefit persists at the parameter scales of contemporary federated-learning deployments, where the geometry of the loss landscape may differ qualitatively from the small-model regime studied here. Third, the application of CosScaffold to non-image domains, including natural-language modelling and tabular classification under federated conditions, would establish whether the directional-modulation principle generalises beyond image-classification tasks. Each of these extensions should preserve the pre-registration discipline applied in the present work, with the comparator set, hyperparameters, and analytical procedures specified before any data is collected.

5.4.4 Cross-Device Federated Settings and Real-World Deployment

The experiments reported here used a simulated cross-silo environment with 100 clients and a sampling fraction of 20% per round, reflecting the federation scale used in the published Scaffold baseline. The cross-device federated learning regime, in which thousands or millions of intermittently available clients with highly variable data quality and computational capacity participate in each round, presents a substantially different optimisation landscape. In such a setting, the previous-round aggregated update on which the cosine signal depends is updated from a very small fraction of the total client population each round, and the resulting signal-to-noise ratio of the directional signal may be markedly different from that observed at the cross-silo scale. The behaviour of the cosine modulator in the cross-device regime is therefore an open empirical question. A second class of real-world deployment

constraints not captured in the simulation used here includes communication latency, packet loss, variable bandwidth, and stochastic client availability. The implementation of CosScaffold within a deployed federated-learning framework, such as TensorFlow Federated or Flower, and the evaluation of its behaviour under realistic network and availability conditions, would establish whether the analytical communication-round savings reported here translate into the wall-clock savings of practical interest. The associated work on per-phase timing instrumentation, identified as a future-work item in Subsection 4.3.4, falls naturally within this theme.

5.4.5 Formal Convergence Analysis

The present work establishes the empirical performance of CosScaffold but does not provide formal theoretical convergence guarantees for the technique. The identity-clamping property ensures that the convergence guarantees of standard Scaffold are preserved at the boundary conditions, but the convergence behaviour of CosScaffold in the regime where the modulator is non-trivially active (that is, where the cosine signal is informative and the modulator departs from the identity) is not formally analysed. A theoretical analysis that derives convergence rates for CosScaffold under standard smoothness and bounded-dissimilarity assumptions, ideally producing a bound that exposes the dependence of the rate on the modulator hyperparameters (the slope α and the clip bounds η_min and η_max), would substantially strengthen the theoretical foundation of the technique. Such an analysis would also clarify whether the empirical observation that the modulator benefit diminishes with rising similarity is reflected in the theoretical rate, and would in turn suggest principled defaults for the modulator hyperparameters rather than the conservative pre-registered values used here. The development of the corresponding lower-bound analysis, identifying the regime in which the cosine modulator can provably improve on standard Scaffold and the regime in which it cannot, would complete the theoretical characterisation and would constitute a substantial contribution to the federated-optimisation literature in its own right.

REFERENCES

[1] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-Efficient Learning of Deep Networks from Decentralized Data," Apr. 10, 2017, PMLR. Accessed: May 13, 2026. [Online]. Available: https://proceedings.mlr.press/v54/mcmahan17a.html

[2] S. P. Karimireddy, S. Kale, M. Mohri, S. J. Reddi, S. U. Stich, and A. T. Suresh, "Scaffold: Stochastic Controlled Averaging for Federated Learning," Nov. 21, 2020, PMLR. Accessed: May 04, 2026. [Online]. Available: https://proceedings.mlr.press/v119/karimireddy20a.html

[3] Y. Zhao, M. Li, L. Lai, N. Suda, D. Civin, and V. Chandra, "Federated Learning with Non-IID Data," Comput. J., Jul. 2022, doi: 10.48550/arXiv.1806.00582.

[4] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated Optimization in Heterogeneous Networks," Proceedings of Machine Learning and Systems, vol. 2, pp. 429–450, Mar. 2020.

[5] D. Alp, et al., "FEDERATED LEARNING BASED ON DYNAMIC REGULARIZATION".

[6] L. Gao, H. Fu, L. Li, Y. Chen, M. Xu, and C. Z. Xu, "FedDC: Federated Learning with Non-IID Data via Local Drift Decoupling and Correction," Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2022-June, pp. 10102–10111, 2022, doi: 10.1109/CVPR52688.2022.00987.

[7] C. Palihawadana, N. Wiratunga, A. Wijekoon, and H. Kalutarage, "FedSim: Similarity guided model aggregation for Federated Learning," Neurocomputing, vol. 483, pp. 432–445, Apr. 2022, doi: 10.1016/J.NEUCOM.2021.08.141.

[8] T. Zhu, et al., "Byzantine-robust Federated Learning via Cosine Similarity Aggregation," Computer Networks, vol. 254, p. 110730, Dec. 2024, doi: 10.1016/J.COMNET.2024.110730.

[9] W. Chang and T. Zhu, "Gradient-based defense methods for data leakage in vertical federated learning," Comput. Secur., vol. 139, p. 103744, Apr. 2024, doi: 10.1016/J.COSE.2024.103744.

[10] F. Varno, M. Saghayi, L. Rafiee Sevyeri, S. Gupta, S. Matwin, and M. Havaei, "AdaBest: Minimizing Client Drift in Federated Learning via Adaptive Bias

Estimation," Lecture Notes in Computer Science, vol. 13683 LNCS, pp. 710–726, 2022, doi: 10.1007/978-3-031-20050-2_41;ISSUE:ISSUE:DOI.

[11] G. I. Quintana, L. Vancamberg, V. Jugnon, M. Mougeot, and A. Desolneux, "BN-Scaffold: controlling the drift of Batch Normalization statistics in Federated Learning," Oct. 2024, Accessed: May 13, 2026. [Online]. Available: https://arxiv.org/pdf/2410.03281

[12] M. J. Page, et al., "The PRISMA 2020 statement: an updated guideline for reporting systematic reviews," BMJ, vol. 372, Mar. 2021, doi: 10.1136/BMJ.N71.

[13] M. J. Page, et al., "PRISMA 2020 explanation and elaboration: updated guidance and exemplars for reporting systematic reviews," BMJ, vol. 372, Mar. 2021, doi: 10.1136/BMJ.N160.

[14] Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated machine learning: Concept and applications," ACM Trans. Intell. Syst. Technol., vol. 10, no. 2, Jan. 2019, doi: 10.1145/3298981;JOURNAL:JOURNAL:TIST;WGROUP:STRING:ACM.

[15] P. Kairouz, et al., "Advances and open problems in federated learning," Foundations and Trends in Machine Learning, vol. 14, no. 1–2, pp. 1–210, Jun. 2021, doi: 10.1561/2200000083;PAGE:STRING:ARTICLE/CHAPTER.

[16] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated Learning: Challenges, Methods, and Future Directions," IEEE Signal Process. Mag., vol. 37, no. 3, pp. 50–60, May 2020, doi: 10.1109/MSP.2020.2975749.

[17] L. Zhu, Z. Liu, and S. Han, "Deep Leakage from Gradients," Adv. Neural Inf. Process. Syst., vol. 32, 2019.

[18] W. Y. B. Lim, et al., "Federated Learning in Mobile Edge Networks: A Comprehensive Survey," IEEE Communications Surveys and Tutorials, vol. 22, no. 3, pp. 2031–2063, Jul. 2020, doi: 10.1109/COMST.2020.2986024.

[19] V. Mothukuri, R. M. Parizi, S. Pouriyeh, Y. Huang, A. Dehghantanha, and G. Srivastava, "A survey on security and privacy of federated learning," Future Generation Computer Systems, vol. 115, pp. 619–640, Feb. 2021, doi: 10.1016/J.FUTURE.2020.10.007.

[20] N. Rieke, et al., "The future of digital health with federated learning," npj Digital Medicine 2020 3:1, vol. 3, no. 1, pp. 119-, Sep. 2020, doi: 10.1038/s41746-020-00323-1.

[21] H. Zhu, J. Xu, S. Liu, and Y. Jin, “Federated learning on non-IID data: A survey,” Neurocomputing, vol. 465, pp. 371–390, Nov. 2021, doi: 10.1016/J.NEUCOM.2021.07.098.

[22] X. Li, K. Huang, W. Yang, S. Wang, and Z. Zhang, “On the Convergence of FedAvg on Non-IID Data,” 2020.

[23] J. Wang, Q. Liu, H. Liang, G. Joshi, and H. V. Poor, “Tackling the Objective Inconsistency Problem in Heterogeneous Federated Optimization,” Adv. Neural Inf. Process. Syst., vol. 33, pp. 7611–7623, 2020.

[24] H. Robbins and S. Monro, “A Stochastic Approximation Method,” https://doi.org/10.1214/aoms/1177729586, vol. 22, no. 3, pp. 400–407, Sep. 1951, doi: 10.1214/AOMS/1177729586.

[25] L. Bottou, F. E. Curtis, and J. Nocedal, “Optimization Methods for Large-Scale Machine Learning,” https://doi.org/10.1137/16M1080173, vol. 60, no. 2, pp. 223–311, May 2018, doi: 10.1137/16M1080173.

[26] “Accelerating stochastic gradient descent using predictive variance reduction | Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 1.” Accessed: May 13, 2026. [Online]. Available: https://dl.acm.org/doi/10.5555/2999611.2999647

[27] A. Defazio Ambiata, F. Bach, and S. Lacoste-Julien, “SAGA: A Fast Incremental Gradient Method With Support for Non-Strongly Convex Composite Objectives,” Adv. Neural Inf. Process. Syst., vol. 27, 2014.

[28] G. Salton, A. Wong, and C. S. Yang, “A Vector Space Model for Automatic Indexing,” Commun. ACM, vol. 18, no. 11, pp. 613–620, Nov. 1975, doi: 10.1145/361219.361220;PAGE:STRING:ARTICLE/CHAPTER.

[29] C. D. Manning, P. Raghavan, and H. Schütze, “Introduction to Information Retrieval,” Introduction to Information Retrieval, Jul. 2008, doi: 10.1017/CBO9780511809071.

[30] Q. Li, B. He, and D. Song, “Model-Contrastive Federated Learning,” Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 10708–10717, 2021, doi: 10.1109/CVPR46437.2021.01057.

[31] H. Wang, et al., “Federated Learning with Matched Averaging,” ArXiv, p. arXiv:2002.06440, Feb. 2020, doi: 10.48550/ARXIV.2002.06440.

[32] Q. Li, Y. Diao, Q. Chen, and B. He, “Federated Learning on Non-IID Data Silos: An Experimental Study,” Proc. Int. Conf. Data Eng., vol. 2022-May, pp. 965–978, 2022, doi: 10.1109/ICDE53745.2022.00077.

[33] A. Fallah, A. Mokhtari, and A. Ozdaglar, “Personalized Federated Learning with Theoretical Guarantees: A Model-Agnostic Meta-Learning Approach,” Adv. Neural Inf. Process. Syst., vol. 33, pp. 3557–3568, 2020.

[34] C. T. Dinh, N. H. Tran, and T. D. Nguyen, “Personalized Federated Learning with Moreau Envelopes,” Adv. Neural Inf. Process. Syst., vol. 33, pp. 21394–21405, 2020.

[35] T. Li, S. Hu, A. Beirami, and V. Smith, “Ditto: Fair and Robust Federated Learning Through Personalization,” Jul. 01, 2021, PMLR. Accessed: May 13, 2026. [Online]. Available: https://proceedings.mlr.press/v139/li21h.html

[36] “Ensemble distillation for robust model fusion in federated learning | Proceedings of the 34th International Conference on Neural Information Processing Systems.” Accessed: May 13, 2026. [Online]. Available: https://dl.acm.org/doi/10.5555/3495724.3495922

[37] G. Lee, M. Jeong, Y. Shin, S. Bae, and S.-Y. Yun, “Preservation of the Global Knowledge by Not-True Distillation in Federated Learning,” Adv. Neural Inf. Process. Syst., vol. 35, pp. 38461–38474, Dec. 2022.

[38] L. M. Nguyen, J. Liu, K. Scheinberg, and M. Takáč, “SARAH: A Novel Method for Machine Learning Problems Using Stochastic Recursive Gradient,” Jul. 17, 2017, PMLR. Accessed: May 13, 2026. [Online]. Available: https://proceedings.mlr.press/v70/nguyen17b.html

[39] J. J. Moreau, J. Jacques, and M. Fonctions, “Fonctions convexes duales et points proximaux dans un espace hilbertien”, Accessed: May 13, 2026. [Online]. Available: https://hal.science/hal-01867195v1

[40] R. T. Rockafellar, “Monotone Operators and the Proximal Point Algorithm,” https://doi.org/10.1137/0314056, vol. 14, no. 5, pp. 877–898, Jul. 2006, doi: 10.1137/0314056.

[41] Neal Parikh and Stephen Boyd, “Proximal Algorithms.” Accessed: May 13, 2026. [Online]. Available: https://web.stanford.edu/%7Eboyd/papers/pdf/prox_algs.pdf

[42] S. Boyd, et al., “Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers,” Foundations and Trends R in Machine Learning, vol. 3, no. 1, pp. 1–122, 2010, doi: 10.1561/2200000016.

[43] “Model-agnostic meta-learning for fast adaptation of deep networks | Proceedings of the 34th International Conference on Machine Learning - Volume 70.” Accessed: May 13, 2026. [Online]. Available: https://dl.acm.org/doi/10.5555/3305381.3305498

[44] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, “A Simple Framework for Contrastive Learning of Visual Representations,” 2020, Accessed: May 13, 2026. [Online]. Available: https://github.com/google-research/simclr.

[45] G. Hinton, O. Vinyals, and J. Dean, “Distilling the Knowledge in a Neural Network,” Mar. 2015, Accessed: May 13, 2026. [Online]. Available: https://arxiv.org/pdf/1503.02531

[46] L. Mächler, I. Ezhov, S. Shit, and J. C. Paetzold, “FedPIDAvg: A PID Controller Inspired Aggregation Method for Federated Learning,” Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 14092 LNCS, pp. 209–217, 2023, doi: 10.1007/978-3-031-44153-0_20/SAVE-RESEARCH.

[47] J. Wang, V. Tantia, N. Ballas, and M. Rabbat, “SLOWMO: IMPROVING COMMUNICATION-EFFICIENT DISTRIBUTED SGD WITH SLOW MOMENTUM”.

[48] S. J. Reddi, et al., “ADAPTIVE FEDERATED OPTIMIZATION”.

[49] S. P. Karimireddy, et al., “Breaking the centralized barrier for cross-device federated learning,” Adv. Neural Inf. Process. Syst., vol. 34, pp. 28663–28676, Dec. 2021.

[50] H. Zhang, C. Li, W. Dai, J. Zou, and H. Xiong, “Federated Learning Based on Model Discrepancy and Variance Reduction,” IEEE Trans. Neural Netw. Learn. Syst., vol. 36, no. 6, pp. 10407–10421, 2025, doi: 10.1109/TNNLS.2024.3517658.

[51] J. Xue and C. Wang, “FedSAGA: Composite Federated Learning with Inertial Douglas-Rachford Splitting and Variance Reduction Method,” pp. 1–9, Jan. 2026, doi: 10.1109/ICPADS67057.2025.11322895.

[52] T. Murata and T. Suzuki, “Bias-Variance Reduced Local SGD for Less Heterogeneous Federated Learning,” Jul. 01, 2021, PMLR. Accessed: May 13, 2026. [Online]. Available: https://proceedings.mlr.press/v139/murata21a.html

[53] D. Jhunjhunwala, P. Sharma, A. Nagarkatti, and G. Joshi, “Fedvarp: Tackling the variance due to partial client participation in federated learning,” Aug. 17, 2022, PMLR. Accessed: May 13, 2026. [Online]. Available: https://proceedings.mlr.press/v180/jhunjhunwala22a.html

[54] C. T. Dinh, N. H. Tran, T. D. Nguyen, W. Bao, A. Y. Zomaya, and B. B. Zhou, “Federated Learning with Proximal Stochastic Variance Reduced Gradient Algorithms,” vol. 20, doi: 10.1145/3404397.3404457.

[55] M. Crawshaw, Y. Bao, and M. Liu, “Federated Learning with Client Subsampling, Data Heterogeneity, and Unbounded Smoothness: A New Algorithm and Lower Bounds,” Adv. Neural Inf. Process. Syst., vol. 36, pp. 6467–6508, Dec. 2023, Accessed: May 13, 2026. [Online]. Available: https://github.com/MingruiLiu-ML-Lab/episode_plusplus.

[56] K. Oko, S. Akiyama, and T. Suzuki, “Reducing Communication in Nonconvex Federated Learning with a Novel Single-Loop Variance Reduction Method”.

[57] “SCAFFLSA | Proceedings of the 38th International Conference on Neural Information Processing Systems.” Accessed: May 13, 2026. [Online]. Available: https://dl.acm.org/doi/10.5555/3737916.3738363

[58] K. Oko, S. Akiyama, D. Wu, T. Murata, and T. Suzuki, “SILVER: Single-loop variance reduction and application to federated learning,” Jul. 08, 2024, PMLR. Accessed: May 13, 2026. [Online]. Available: https://proceedings.mlr.press/v235/oko24a.html

[59] Y. Liu, X. Yang, X. Chen, H. Du, and L. Xu, “FedNCV: Optimizing Federated Learning With Networked Control Variates,” Transactions on Emerging Telecommunications Technologies, vol. 36, no. 11, p. e70287, Nov. 2025, doi: 10.1002/ETT.70287;ISSUE:ISSUE:DOI.

[60] O. Mashaal, S. Baadel, and H. Abouzied, “Extending Control Theory into Federated Learning Data Heterogeneity Problem,” International Conference on Artificial Intelligence and Mechatronics System, AIMS 2024, 2024, doi: 10.1109/AIMS61812.2024.10512401.

[61] R. Dai, et al., “FedGAMMA: Federated Learning With Global Sharpness-Aware Minimization,” IEEE Trans. Neural Netw. Learn. Syst., vol. 35, no. 12, pp. 17479–17492, 2024, doi: 10.1109/TNNLS.2023.3304453.

[62] X. Feng, M. P. Laiu, and T. Strohmer, “FedOSAA: Improving Federated Learning with One-Step Anderson Acceleration,” Jun. 15, 2025, PMLR. Accessed: May 13, 2026. [Online]. Available: https://proceedings.mlr.press/v280/feng25a.html

[63] Y. Xu, W. Ma, C. Dai, Y. Wu, and H. Zhou, “Generalized Federated Learning via Gradient Norm-Aware Minimization and Control Variables,” Mathematics 2024, Vol. 12, Page 2644, vol. 12, no. 17, p. 2644, Aug. 2024, doi: 10.3390/MATH12172644.

[64] Z. Cheng, X. Huang, P. Wu, and K. Yuan, “Momentum Benefits Non-iid Federated Learning Simply and Provably,” International Conference on Learning Representations, vol. 2024, pp. 9815–9848, May 2024.

[65] S. Wang, Y. Xu, Z. Wang, T. H. Chang, T. Q. S. Quek, and D. Sun, “Beyond ADMM: A Unified Client-Variance-Reduced Adaptive Federated Learning Framework,” Proceedings of the 37th AAAI Conference on Artificial Intelligence, AAAI 2023, vol. 37, pp. 10175–10183, Jun. 2023, doi: 10.1609/AAAI.V37I8.26212;TAXONOMY:TAXONOMY:ACM-PUBTYPE;PAGEGROUP:STRING:PUBLICATION.

[66] W. Yan, K. Zhang, X. Wang, and X. Cao, “Problem-Parameter-Free Federated Learning,” International Conference on Learning Representations, vol. 2025, pp. 97350–97377, May 2025.

[67] A. Authors, “SCAFF-PD: Communication Efficient Fair and Robust Federated Learning”.

[68] X. Wang, L. Tian, C. Yang, F. Lin, and M. Li, “Enhancing Federated Learning under Partial Participation via Proportional Variance Reduction,” Proceedings of the IEEE International Conference on Trust, Security and Privacy in Computing and Communications, TrustCom, no. 2025, pp. 344–351, 2025, doi: 10.1109/TRUSTCOM66490.2025.00045.

[69] M. Crawshaw and M. Liu, “Federated Learning under Periodic Client Participation and Heterogeneous Data: A New Communication-Efficient Algorithm and Analysis,” Oct. 2024, Accessed: May 13, 2026. [Online]. Available: https://arxiv.org/pdf/2410.23131

[70] B. Li, M. N. Schmidt, T. S. Alstrom, and S. U. Stich, “On the Effectiveness of Partial Variance Reduction in Federated Learning with Heterogeneous Data,” Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2023-June, pp. 3964–3973, 2023, doi: 10.1109/CVPR52729.2023.00386.

[71] I. Iwan, B. N. Yahya, and S. L. Lee, “Federated model with contrastive learning and adaptive control variates for human activity recognition,” Frontiers of Information Technology & Electronic Engineering 2025 26:6, vol. 26, no. 6, pp. 896–911, Jul. 2025, doi: 10.1631/FITEE.2400797.

[72] N. Moneesh, S. Madhan, D. G. Nair, and J. J. Nair, “FedHybrid: Unifying Aggregation Strategies to Optimize Federated Learning on Non-IID Dataset,” Procedia Comput. Sci., vol. 258, pp. 3126–3134, Jan. 2025, doi: 10.1016/J.PROCS.2025.04.570.

[73] Q. Zhang, et al., “Robust Federated Fuzzy C-Means Algorithm in Heterogeneous Scenarios,” IEEE Transactions on Fuzzy Systems, vol. 33, no. 9, pp. 3168–3181, 2025, doi: 10.1109/TFUZZ.2025.3584697.

[74] W. T. Chang, M. Seif, and R. Tandon, “Differentially Private Federated Learning with Drift Control,” 2022 56th Annual Conference on Information Sciences and Systems, CISS 2022, pp. 240–245, 2022, doi: 10.1109/CISS53076.2022.9751200.

[75] W. Ding, H. Guo, Z. Yan, and M. Wang, “Secure Federated Learning Schemes Based on Multi-Key Homomorphic Encryption,” Proceedings of the IEEE International Conference on Trust, Security and Privacy in Computing and Communications, TrustCom, no. 2024, pp. 551–558, 2024, doi: 10.1109/TRUSTCOM63139.2024.00091.

[76] T. Gafni, K. Cohen, and Y. C. Eldar, “Federated Learning from Heterogeneous Data via Controlled Air Aggregation with Bayesian Estimation,” IEEE Transactions on Signal Processing, vol. 72, pp. 1928–1943, 2024, doi: 10.1109/TSP.2024.3351469.

[77] X. Huang, P. Li, and X. Li, “Stochastic Controlled Averaging for Federated Learning with Communication Compression,” International Conference on Learning Representations, vol. 2024, pp. 27956–27994, May 2024.

[78] M. G. Rahmat and M. Khalilian, “A Novel Pearson Correlation-Based Merging Algorithm for Robust Distributed Machine Learning with Heterogeneous Data,” Jan. 2025, Accessed: May 13, 2026. [Online]. Available: https://arxiv.org/pdf/2501.11112

[79] H. Yang, Z. Liu, X. Zhang, and J. Liu, “SAGDA: Achieving \mathcal{O}(\epsilon^{-2}) Communication Complexity in Federated Min-Max Learning,” Adv. Neural Inf. Process. Syst., vol. 35, pp. 7142–7154, Dec. 2022.

[80] P. Mangold, A. O. Durmus, A. Dieuleveut, and E. Moulines, “Scaffold with Stochastic Gradients: New Analysis with Linear Speed-Up,” Oct. 06, 2025, PMLR.

Accessed: May 13, 2026. [Online]. Available: https://proceedings.mlr.press/v267/mangold25a.html

[81] J. Konečn, H. Brendan McMahan, F. X. Yu, A. Theertha Suresh, D. Bacon Google, and P. Richtárik, "Federated Learning: Strategies for Improving Communication Efficiency," Oct. 2016, Accessed: May 13, 2026. [Online]. Available: https://arxiv.org/pdf/1610.05492

[82] J. Wang, R. Das, G. Joshi, S. Kale, Z. Xu, and T. Zhang, "On the Unreasonable Effectiveness of Federated Averaging with Heterogeneous Data," Transactions on Machine Learning Research, vol. 2024, 2024, Accessed: May 13, 2026. [Online]. Available: https://experts.illinois.edu/en/publications/on-the-unreasonable-effectiveness-of-federated-averaging-with-het/

[83] F. Sattler, K. R. Muller, and W. Samek, "Clustered Federated Learning: Model-Agnostic Distributed Multitask Optimization under Privacy Constraints," IEEE Trans. Neural Netw. Learn. Syst., vol. 32, no. 8, pp. 3710–3722, Aug. 2021, doi: 10.1109/TNNLS.2020.3015958.

[84] T.-M. H. Hsu, H. Qi, G. Research, and M. Brown, "Measuring the Effects of Non-Identical Data Distribution for Federated Visual Classification," Sep. 2019, Accessed: May 13, 2026. [Online]. Available: https://arxiv.org/pdf/1909.06335

[85] T. Li, M. Sanjabi, A. Beirami, and V. Smith, "Fair Resource Allocation in Federated Learning," 8th International Conference on Learning Representations, ICLR 2020, May 2019, Accessed: May 13, 2026. [Online]. Available: https://arxiv.org/pdf/1905.10497

[86] M. Mohri, G. Sivek, and A. T. Suresh, "Agnostic Federated Learning," May 24, 2019, PMLR. Accessed: May 13, 2026. [Online]. Available: https://proceedings.mlr.press/v97/mohri19a.html

[87] R. Jain, D. Chiu, and W. Hawe, "A Quantitative Measure Of Fairness And Discrimination For Resource Allocation In Shared Computer Systems," Sep. 1998, Accessed: May 14, 2026. [Online]. Available: https://arxiv.org/pdf/cs/9809099

[88] S. Caldas, et al., "LEAF: A Benchmark for Federated Settings," Dec. 2018, Accessed: May 13, 2026. [Online]. Available: https://arxiv.org/pdf/1812.01097

[89] F. Lai, et al., "FedScale: Benchmarking Model and System Performance of Federated Learning at Scale," Jun. 28, 2022, PMLR. Accessed: May 13, 2026. [Online]. Available: https://proceedings.mlr.press/v162/lai22a.html

[90] “A Simple Sequentially Rejective Multiple Test Procedure on JSTOR.” Accessed: May 15, 2026. [Online]. Available: https://www.jstor.org/stable/4615733?seq=1

[91] G. Cohen, S. Afshar, J. Tapson, and A. Van Schaik, “EMNIST: Extending MNIST to handwritten letters,” Proceedings of the International Joint Conference on Neural Networks, vol. 2017-May, pp. 2921–2926, Jun. 2017, doi: 10.1109/IJCNN.2017.7966217.

[92] A. Krizhevsky, “Learning Multiple Layers of Features from Tiny Images,” 2009.

[93] “Bootstrap Methods: Another Look at the Jackknife on JSTOR.” Accessed: May 15, 2026. [Online]. Available: https://www.jstor.org/stable/2958830

[94] F. Wilcoxon, “Individual Comparisons by Ranking Methods,” Biometrics Bulletin, vol. 1, no. 6, p. 80, Dec. 1945, doi: 10.2307/3001968.

[95] S. S. Shapiro and M. B. Wilk, “An Analysis of Variance Test for Normality (Complete Samples),” Biometrika, vol. 52, no. 3/4, p. 591, Dec. 1965, doi: 10.2307/2333709.

[96] A. Paszke, et al., “PyTorch: An Imperative Style, High-Performance Deep Learning Library,” Adv. Neural Inf. Process. Syst., vol. 32, 2019.

[97] M. Yurochkin, M. Agarwal, S. Ghosh, K. Greenewald, T. N. Hoang, and Y. Khazaeni, “Bayesian Nonparametric Federated Learning of Neural Networks,” May 24, 2019, PMLR. Accessed: May 15, 2026. [Online]. Available: https://proceedings.mlr.press/v97/yurochkin19a.html

[98] A. Defazio and L. Bottou, “On the Ineffectiveness of Variance Reduced Optimization for Deep Learning,” in Advances in Neural Information Processing Systems 32 (NeurIPS 2019)

[99] A. Hard, et al., “Federated Learning for Mobile Keyboard Prediction,” Nov. 2018, Accessed: May 15, 2026. [Online]. Available: https://arxiv.org/pdf/1811.03604

APPENDICES

Appendix 1: Questionnaire

Expert Opinion Survey

This survey aims to obtain your expert opinion on the potential of extending SCAFFOLD's control variate mechanism using cosine similarity for client drift mitigation in Federated Learning.

Your responses will help validate findings and recommendations from a systematic literature review on approaches employed in extending SCAFFOLD for client drift mitigation. The survey will take approximately 10-15 minutes to complete. All responses will be treated with the utmost confidentiality and anonymity.

Section A: Demographic Information

1. Role/Sector

☐ Academic/Researcher

☐ Industry Practitioner

☐ Postgraduate Student

☐ Other: _____

2. Highest Academic Level

☐ First Degree

☐ Masters

☐ PHD

☐ Other: _____

3. Experience with Federated Learning

☐ None

☐ Basic

☐ Intermediate

☐ Advanced

☐ Expert

4. Experience with Scaffold or Control Variate techniques

☐ None

☐ Basic

☐ Intermediate

☐ Advanced

☐ Expert

5. Years of experience in Machine Learning or distributed Systems?

☐ Less than 1 year

☐ 1 - 3 years

☐ 4 - 7 years

☐ 8 - 10 years

☐ More than 10 years

Section B: Expert Opinion on SCAFFOLD Extension via Cosine Similarity

In this section, please indicate your level of agreement with each statement using the five-point Likert scale provided.

Parameters Identified from SLR are:

Statement	**Strongly Disagree**	**Disagree**	**Neutral**	**Agree**	**Strongly Agree**
Client drift caused by heterogeneous (non-IID) data across clients is a significant challenge affecting the convergence and performance of Federated Learning systems.	☐	☐	☐	☐	☐
SCAFFOLD's existing control variate mechanism (server and client control variates) effectively mitigates client drift under moderate data heterogeneity.	☐	☐	☐	☐	☐
SCAFFOLD's drift mitigation performance degrades significantly when data heterogeneity among clients is extreme.	☐	☐	☐	☐	☐
The communication bottleneck imposed by	☐	☐	☐	☐	☐

transmitting additional control variates hinders SCAFFOLD's deployment in bandwidth-constrained Federated Learning settings.					
Similarity measures that evaluate the directional alignment between client and global gradients can serve as meaningful indicators of client drift severity.	☐	☐	☐	☐	☐
A similarity-based approach that proactively corrects divergence before applying model updates is more effective than approaches treating drift after it has occurred.	☐	☐	☐	☐	☐
Cosine similarity, which measures directional alignment between vectors irrespective of their magnitude, is a suitable metric for identifying the alignment between local client gradients and global gradients in Federated Learning.	☐	☐	☐	☐	☐
Cosine similarity offers computational advantages over other similarity measures for gradient-based drift detection, particularly in high-dimensional parameter spaces.	☐	☐	☐	☐	☐
Incorporating cosine similarity as a third control variate alongside the	☐	☐	☐	☐	☐

existing server and client control variates is a technically feasible extension of the SCAFFOLD protocol.					
A similarity-based control variate that modulates drift correction dynamically based on divergence severity would improve upon SCAFFOLD's uniform correction strategy.	☐	☐	☐	☐	☐
An adaptive, cosine similarity-driven correction would not introduce unnecessary computational overhead compared to existing SCAFFOLD extensions.	☐	☐	☐	☐	☐
A cosine similarity-based control variate would improve SCAFFOLD's robustness to extreme non-IID data distributions without imposing restrictive theoretical assumptions (e.g., strong convexity or Gaussian distributions).	☐	☐	☐	☐	☐
Investigating cosine similarity as a third control variate for client drift mitigation is a worthwhile direction for future Federated Learning research.	☐	☐	☐	☐	☐

Section C: Additional Comments

6. In your expert opinion, what other similarity measures (if any) could be considered alongside or instead of cosine similarity for client drift mitigation in SCAFFOLD?

_____

_____

_____

_____

_____.

7. Are there any additional challenges or considerations you foresee in incorporating a similarity-based control variate into the SCAFFOLD protocol?

_____

_____

_____

_____

_____.

Appendix 2: BPS Letter

MURANG'A UNIVERSITY OF TECHNOLOGY

Office of the Director, Postgraduate Studies

E-mail: bps@mut.ac.ke
Website: www.mut.ac.ke

P.O. Box 75-10200
Murang'a, Kenya

Ref: MUT/ARP/PGS/19/2020 /VOL.1 **Date:** 9th February, 2026

MR. JAMES MBURU MUTHII-SC400/5143/2023

School of Computing and Information Technology,
Murang'a University of Technology,
P.O. Box 75-10,200
Murang'a.

Dear Mr. Muthii,

RE: APPROVAL OF RESEARCH PROPOSAL AND SUPERVISORS.

I am pleased to inform you that the Directorate of Postgraduate Studies on 28th January, 2026 considered and approved your Masters research proposal entitled ***"An Extended Scaffold Technique Based on Cosine Similarity for Client Drift Mitigation in Federated Learning"*** and appointed the following as supervisors:

1. Dr. Stephen Kahara – Murang'a University of Technology
2. Dr. Stephen Njenga – Murang'a University of Technology

You may now proceed with your data collection subject to obtaining research permit from NACOSTI, if required. You should also begin consulting your supervisors and submit through them quarterly progress reports to the Director Postgraduate Studies through your CoD and School Dean. Progress Reports forms can be accessed in the Student Portal.

It is the policy and regulations of the University that you observe deadlines. The guidelines on Postgraduate supervision can be accessed in the post graduate Handbook.

Your responsibilities as a student will include, among others;

i. Maintain regular consultation with your supervisor(s), at least once a month
ii. Submit quarterly reports on time, through your supervisors, CoD, Dean and to the Director of Postgraduate Studies;

iii. Ensure quality work all through;
iv. Present your research findings at 2-3 seminars/conferences prior to thesis examination.
v. Publish one article from your research findings in a refereed journal prior to thesis examination.

For any further clarification please contact the undersigned.

Yours Sincerely,

DIRECTORATE OF POSTGRADUATE STUDIES
MURANG'A UNIVERSITY OF TECHNOLOGY
09 FEB 2026
P.O. Box 75 - 10200, MURANG'A
Tel: 0771463515, Email: info@mut.ac.ke

Prof. Geoffrey Muchiri, PhD
Director, Postgraduate Studies
GM/MMK

Copy to:
1. *Registrar – Academics and Students Affairs*
2. Dean, School of Computing and Information Technology

Appendix 3: Institutional Scientific Ethics Review Committee (ISERC) Approval

OFFICE OF INSTITUTIONAL SCIENTIFIC ETHICS REVIEW COMMITTEE
MURANG'A UNIVERSITY OF TECHNOLOGY
Website: www.mut.ac.ke ; Email: iserc@mut.ac.ke

March 5, 2026

National Commission for Science Technology & Innovation (NACOSTI)
Off Waiyaki Way, Upper Kabete
P.O. Box 30623-00100
NAIROBI, KENYA.

Dear Sir/Madam,

RE: JAMES MBURU MUTHII, Reg.No: SC400/5143/2023

The purpose of this letter is to introduce the above named student who is pursuing Degree of Master of Science in Computer Science of Murang'a University of Technology.

The title of the research is *"An Extended Scaffold Technique Based on Cosine Similarity for Client Drift Mitigation in Federated Learning"*.

The research proposal has been cleared by the University's MUT-ISERC Committee (approval letter attached) and now has to proceed to the field to collect data between 5th March 2026 and 4th March 2027.

Any assistance accorded to the student will be highly appreciated.

Thank you.

Yours sincerely

INSTITUTIONAL SCIENTIFIC ETHICS REVIEW COMMITTEE
MURANG'A UNIVERSITY OF TECHNOLOGY
05 MAR 2026
CHAIRMAN

Name Dr. Dennis Otieno, PhD.
CHAIRMAN- MUT ISERC

Appendix 4: NACOSTI Research License

REPUBLIC OF KENYA

NATIONAL COMMISSION FOR SCIENCE,TECHNOLOGY & INNOVATION

Ref No: **430001**

Date of Issue: **06/March/2026**

RESEARCH LICENSE



This is to Certify that Mr.. JAMES MUTHII MBURU of Murang'a University of Technology, has been licensed to conduct research as per the provision of the Science, Technology and Innovation Act, 2013 (Rev.2014) in Muranga on the topic: AN EXTENDED SCAFFOLD TECHNIQUE BASED ON COSINE SIMILARITY FOR CLIENT DRIFT MITIGATION IN FEDERATED LEARNING for the period ending : 06/March/2027.

License No: **NACOSTI/P/26/4186274**

430001

Applicant Identification Number

Ag. Director General
NATIONAL COMMISSION FOR SCIENCE, TECHNOLOGY & INNOVATION

Verification QR Code

NOTE: This is a computer generated License. To verify the authenticity of this document, Scan the QR Code using QR scanner application.

See overleaf for conditions

Appendix 5: Publication

This thesis has produced the following publication:

[1] J. M. Muthii, Dr. S. K. Wanjau, and Dr. S. Njenga, ‘Scaffold Extensions for Client Drift Mitigation in Federated Learning: A Synthesis of Approaches, Limitations, and Future Directions’, IJESE, vol. 14, no. 4, pp. 1–12, Mar. 2026, doi: 10.35940/ijese.d2641.14040326.