ONLINE ASYNCHRONOUS FEDERATED CONFORMAL PREDICTION

Anonymous authors
Paper under double-blind review

ABSTRACT

We propose OAFCP, a novel online asynchronous federated conformal prediction framework for uncertainty quantification across heterogeneous clients with streaming data. Specifically, our method jointly updates model parameters and conformal quantiles as new observations arrive, producing adaptive prediction intervals for regression and prediction sets for classification without sharing raw data. Clients process their data streams at different speeds, while the server aggregates available updates at fixed intervals without waiting for all clients. The aggregation weights compensate for differences in computation speed to balance client contributions and ensure long-term aggregation fairness. On the theoretical side, we establish long-run coverage guarantees under asynchronous aggregation and, under score-distribution stability conditions, show that client-weighted instantaneous coverage converges to the nominal level. Experiments on synthetic data and six real-world datasets demonstrate the finite-sample performance of OAFCP.

1 INTRODUCTION

As data become increasingly distributed across institutions and devices, federated learning (FL) enables collaborative model training without centralizing raw observations (McMahan et al., 2017; Li et al., 2020a; 2026). In health monitoring, for example, hospitals, smartphones, and wearable devices can contribute to a shared predictive model by exchanging locally computed updates while keeping their measurements local. Such collaboration, however, must accommodate differences in computing capacity, network connectivity, and device availability. In synchronous methods such as FedAvg, these differences can cause slower clients to delay global aggregation, leaving faster clients idle (McMahan et al., 2017; Li et al., 2020b). Although FedProx and Scaffold methods mitigate optimization challenges arising from heterogeneous data and local computation, they retain round-based synchronization (Li et al., 2020b; Karimireddy et al., 2020). To mitigate this synchronization bottleneck, asynchronous federated learning (AFL) allows the server to incorporate client updates as they become available, without waiting for all participating clients to complete their local computations; see, for example, (Xie et al., 2019; Nguyen et al., 2022; Fraboni et al., 2023; Wang et al., 2024; Leconte et al., 2024; Yan et al., 2025). Nevertheless, AFL accommodates heterogeneity in computation, communication, and client participation, most existing methods assume that each client's local dataset is fixed before training begins.

The need to continuously incorporate newly arriving observations has motivated online federated learning, in which model parameters are updated sequentially as observations become available. Marfoq et al. (2023) developed a weighted empirical risk minimization FL framework that incorporates observations collected during training. At the algorithmic level, local SGD (Stich, 2019) combined parallel stochastic gradient updates at each client with periodic model averaging at the server, thereby reducing communication while retaining recursive local learning (Gorbunov et al., 2021; Li et al., 2022). Building on this paradigm, Hong & Chae (2022) developed a communication-efficient online multiple-kernel learning method with sublinear regret, while Ghari & Shen (2022) proposed personalized online kernel predictors to accommodate heterogeneous client distributions. Addressing asynchronous execution, Gauthier et al. (2023) introduced a partial-sharing framework that allows for intermittent participation and delayed client updates. Beyond optimization-oriented analyses, a growing body of work has begun to investigate the statistical properties of FL. Bao et al. (2022) studied statistical recovery in federated settings, while Li et al. (2022) developed online inference theory for local SGD. Such parameter-level guarantees, however, do not directly provide

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

predictive uncertainty quantification in streaming federated environments, where observations arrive sequentially, client distributions may differ, and model updates are aggregated asynchronously.

Conformal prediction (CP) (Vovk et al., 2005) provides a distribution-free framework for constructing prediction sets with finite-sample coverage guarantees. An early extension of CP to sequential settings was developed by Gibbs & Candès (2021), who used online convex optimization to adapt prediction thresholds in response to observed coverage errors. Building on this sequential calibration paradigm, Angelopoulos et al. (2024) introduced decaying step sizes that maintain retrospective coverage control while improving quantile estimation under suitable stationarity conditions. Subsequent work has explored feedback-control methods (Angelopoulos et al., 2023), strongly adaptive online learning (Bhatnagar et al., 2023), Bayesian formulations (Zhang et al., 2024), privacy-preserving mechanisms (Zhang et al., 2025a;b), and parameter-free portfolio methods (Liu et al., 2026; Bharti et al., 2026). Although these developments enable prediction sets to adapt over time, they largely focus on centralized settings and do not address how calibration should be coordinated across heterogeneous federated clients with asynchronously arriving updates.

To address this gap, we introduce **OAFCP**, a unified framework for predictive uncertainty quantification over asynchronous federated data streams. As illustrated in Figure 1, clients process newly arriving observations and transmit local model and conformal-threshold updates at heterogeneous rates, while the server aggregates the updates available at each round without waiting for all clients. Our main contributions are summarized as follows.

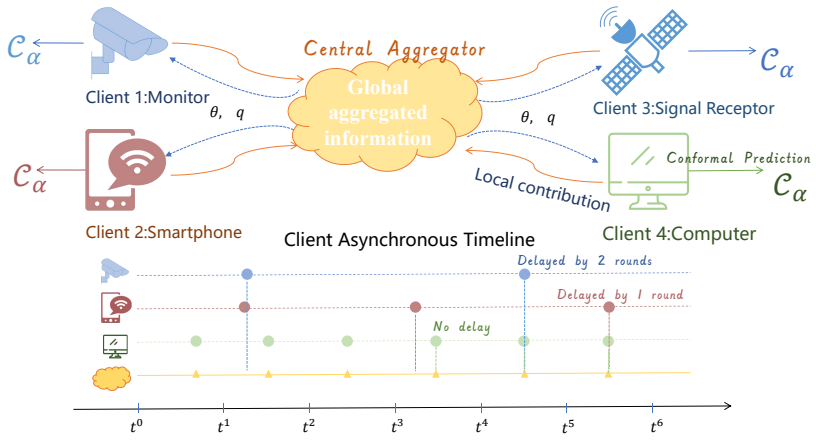


Figure 1: Overview of OAFCP. Clients asynchronously process local data streams and transmit model and conformal-threshold updates. At each server round, the server aggregates the available updates without waiting for all clients. The lower panel illustrates heterogeneous client delays.

- **Online asynchronous federated conformal prediction.** We propose OAFCP, which, to the best of our knowledge, is the first framework for predictive uncertainty quantification over asynchronous federated data streams. OAFCP jointly updates predictive models and conformal calibration thresholds as observations arrive, supporting both regression and classification in heterogeneous federated environments.
- **Coverage guarantees.** We establish long-run coverage guarantees for OAFCP and, under suitable score-distribution stability conditions, prove almost sure convergence of the global and local calibration thresholds to a common target quantile, thereby yielding asymptotic client-weighted instantaneous coverage at the nominal level.
- **Comprehensive empirical evaluation.** We conduct comprehensive experiments on synthetic data and six real-world datasets. OAFCP achieves competitive predictive performance, tighter prediction intervals with coverage closer to the target on regression, performance comparable to the synchronous baseline on classification, and robustness to abrupt and gradual distribution shifts in heterogeneous asynchronous settings.

2 PROBLEM FORMULATION

In this section, we consider an online FL framework with M clients and a central server. Clients collaboratively update a predictive model $f_{\boldsymbol{\theta}}$, parameterized by $\boldsymbol{\theta} \in \mathbb{R}^d$, from local data streams without sharing raw observations. Differences in data arrival rates, computational resources, and communication speeds lead to unequal update frequencies and communication delays. The server therefore aggregates available model and calibration updates asynchronously, without requiring synchronized client participation. Specifically, each client $i \in \{1, \ldots, M\}$ receives observations sequentially: $\{(\boldsymbol{x}_t^i, y_t^i)\}_{t\geq 1},\ (\boldsymbol{x}_t^i, y_t^i) \sim \mathcal{D}_i^t$, where t indexes observations within the client's stream rather than communication rounds. Here, $\boldsymbol{x}_t^i \in \mathcal{X}$ is the input feature, $y_t^i \in \mathcal{Y}$ is the response, and $\mathcal{D}_i^t$ is their joint distribution. For regression, $\mathcal{Y} \subseteq \mathbb{R}$; for classification, $\mathcal{Y}$ is a finite label set. The distributions may differ across clients and evolve over time. Each client processes observations sequentially, maintaining its model and calibration state without re-accessing raw historical data.

Traditional FL focuses on model estimation by minimizing a weighted population risk (Li et al., 2022; Fraboni et al., 2023):

$$\min_{\boldsymbol{\theta}\in\mathbb{R}^d} \mathcal{L}(\boldsymbol{\theta}) = \sum_{i=1}^{M} p_i \mathcal{L}_i(\boldsymbol{\theta}) := \sum_{i=1}^{M} p_i \mathbb{E}_{(\boldsymbol{x}_t^i, y_t^i)\sim\mathcal{D}_i^t} \left\{ \ell(\boldsymbol{\theta}; \boldsymbol{x}_t^i, y_t^i) \right\} \tag{1}$$

where $\ell(\cdot)$ is the loss function and the important weights of these clients satisfy $p_i \geq 0$ and $\sum_{i=1}^{M} p_i = 1$. The objective function in Equation (1) guides model learning but does not, by itself, provide coverage guarantees for prediction sets. To quantify predictive uncertainty, we review online conformal prediction, which calibrates prediction sets using sequential error feedback (Gibbs & Candès, 2021; Bhatnagar et al., 2023; Angelopoulos et al., 2024).

Let $\boldsymbol{\theta}_t^i$ denote the model parameters available to client i at its tth prediction, and let $s_t^i : \mathcal{X} \times \mathcal{Y} \to [0, B]$, with $B < \infty$, be the associated nonconformity score. Larger scores indicate greater disagreement with the model. For regression, a common choice is the absolute residual score, $s_t^i(\boldsymbol{x}, y) = |y - f_{\boldsymbol{\theta}_t^i}(\boldsymbol{x})|$. For classification, one may use the probability-based score $s_t^i(\boldsymbol{x}, y) = 1 - [f_{\boldsymbol{\theta}_t^i}(\boldsymbol{x})]_y$, where $[f_{\boldsymbol{\theta}_t^i}(x)]_y$ denotes the predicted probability of class y. This score lies in $[0, 1]$ and assigns larger values to labels considered less likely by the model. Given a prescribed miscoverage level $\alpha \in (0, 1)$, client i observes $\boldsymbol{x}_t^i$ and constructs $C_t^i := \left\{ y \in \mathcal{Y} : s_t^i(\boldsymbol{x}_t^i, y) \leq q_t^i \right\}$, where the score function and threshold $q_t^i \in \mathbb{R}$ are determined before y_t^i is revealed. Once the response is observed, the client computes the realized score and miscoverage indicator,

$$\mathrm{err}_t^i := \mathbf{1}\{y_t^i \notin C_t^i\} = \mathbf{1}\{S_t^i > q_t^i\} \text{ with } S_t^i := s_t^i(\boldsymbol{x}_t^i, y_t^i).$$

This feedback drives threshold adaptation through subgradient descent on the pinball loss (Bhatnagar et al., 2023; Angelopoulos et al., 2024), $\rho_{1-\alpha}(q; S_t^i) := (S_t^i - q)\left(\mathbf{1}\{S_t^i > q\} - \alpha\right)$. Using the valid subgradient $\alpha - \mathrm{err}_t^i$ at $q = q_t^i$ yields the local update

$$q_{t+1}^i = q_t^i - \eta_t g_t^i, \text{ with } g_t^i := \alpha - \mathrm{err}_t^i \in \partial_q \rho_{1-\alpha}(q_t^i; S_t^i),$$

where $\eta_t > 0$ is the step size and q_{t+1}^i is the locally updated threshold before server aggregation. The corresponding threshold increases after a miscoverage event and decreases otherwise.

In an asynchronous federated framework, however, calibration updates arrive at different rates and may reflect heterogeneous client distributions or stale model states. Server aggregation also modifies local thresholds, so the resulting calibration sequence no longer follows the standalone update above. Its coverage guarantees therefore require accounting for unequal participation and delayed communication. To formalize this problem, let $n_i(T)$ denote the number of completed prediction evaluation steps at client i by system time T, and let $K_T := \sum_{i=1}^{M} n_i(T)$. We evaluate pooled coverage over all completed predictions, assigning equal weight to each prediction rather than each client. Our primary goal is to develop an asynchronous online federated conformal prediction framework that produces informative prediction sets while achieving the long-run coverage objective:

$$\left| \frac{1}{K_T} \sum_{i=1}^{M} \sum_{t=1}^{n_i(T)} \mathbf{1}\{y_t^i \in C_t^i\} - (1-\alpha) \right| \longrightarrow 0 \ \text{ as } \ K_T \to \infty.$$

3 METHODOLOGY

In this section, we present OAFCP, a framework for online uncertainty quantification in asynchronous federated learning. Our method constructs online prediction sets while accommodating heterogeneous client update rates and communication delays. Figure 2 illustrates the client–server interactions and fixed-interval aggregation procedure.

3.1 TIMING AND ASYNCHRONICITY

Each local client processes K observations sequentially, updating its predictive model in an online manner and performing one conformal calibration step per observation. Let $\tau_i > 0$ denote the time required for client i to complete a task and deliver its update to the server. We assume that τ_i is fixed for each client but may differ across clients. Without loss of generality, clients are ordered such that $\tau_{\min} = \tau_1 \leq \tau_2 \leq \cdots \leq \tau_M = \tau_{\max}$. Synchronous aggregation requires waiting for slower clients, whereas aggregation upon every arriving update can lead to frequent server communication. To balance these costs, we use the fixed-interval aggregation policy of FedFix (Fraboni et al., 2023). Let $\Delta_t > 0$ denote the fixed time interval between consecutive server aggregations. With initialization at $t^0 = 0$, the server aggregates available client updates at $t^n = n\Delta_t$, $n = 1, 2, \ldots$. Let $\boldsymbol{\theta}^n$ and q^n denote the global model parameters and calibration threshold at time t^n. Server round n spans $[t^n, t^{n+1})$ and ends with the aggregation that produces $(\boldsymbol{\theta}^{n+1}, q^{n+1})$. Here, n indexes server rounds, whereas the subscript t in $\boldsymbol{\theta}_t^i$ and q_t^i continues to index observations within client i's stream.

Specifically, all clients initialize their first local task from $(\boldsymbol{\theta}^0, q^0)$ at time $t^0 = 0$. At the start of server round n, let $\rho_i(n)$ the index of the global model–threshold pair used to initialize client i's current task. This task began at time $t^{\rho_i(n)}$ from $(\boldsymbol{\theta}^{\rho_i(n)}, q^{\rho_i(n)})$, with $\rho_i(0) = 0$ and $0 \leq \rho_i(n) \leq n$. The difference $n - \rho_i(n)$ measures the staleness of this initialization in server rounds. At time t^n, the remaining time until client i's update becomes available to the server is

$$T_i^n := \tau_i - (t^n - t^{\rho_i(n)}) = \tau_i - (n - \rho_i(n))\,\Delta_t.$$

Client i therefore contributes to the aggregation at t^{n+1} if and only if $T_i^n \leq \Delta_t$. Accordingly, the participating set for round n is $\mathcal{S}_n := \{i \in \{1, \ldots, M\} : T_i^n \leq \Delta_t\}$.

At time t^{n+1}, the server aggregates the model and calibration updates from clients in $\mathcal{S}_n$ to obtain $(\boldsymbol{\theta}^{n+1}, q^{n+1})$. Participating clients receive this global state and begin their next local task, while nonparticipating clients continue their unfinished tasks without restarting. Consequently,

$$\rho_i(n+1) = \begin{cases} n+1, & i \in \mathcal{S}_n, \\ \rho_i(n), & i \notin \mathcal{S}_n. \end{cases}$$

Clients that finish early wait until the next aggregation before restarting, so each client contributes at most one update per server round.

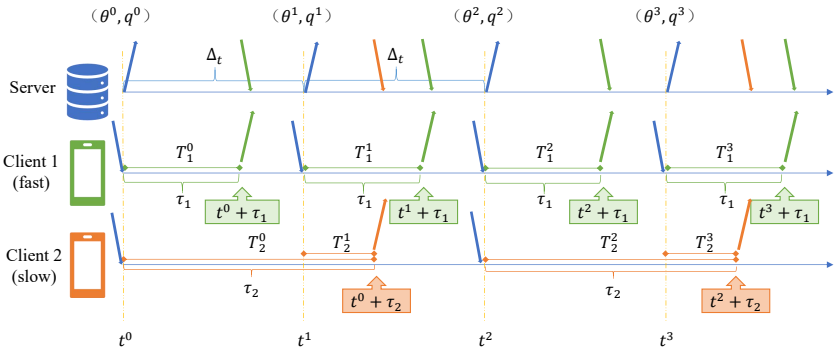


Figure 2: Timing and asynchronous aggregation in OAFCP. The server aggregates every Δ_t time units; τ_i denotes the task and delivery duration, and T_i^n the remaining time at t^n. Clients with $T_i^n \leq \Delta_t$ contribute at t^{n+1} and restart from $(\boldsymbol{\theta}^{n+1}, q^{n+1})$; others continue their tasks.

3.2 Local Client Updates

For each client $i \in \mathcal{S}_n$, consider the local task contributing to the aggregation at time t^{n+1}. The task begins at time $t^{\rho_i(n)}$ from the global state $(\boldsymbol{\theta}^{\rho_i(n)}, q^{\rho_i(n)})$ and processes K observations sequentially, performing one model update and one conformal calibration step per observation. Using the observation count $n_i(T)$ defined in Section 2, the kth observation in this task is denoted by $(\boldsymbol{x}_t^i, y_t^i)$, where $t = n_i(t^{\rho_i(n)}) + k,\ k = 1, \ldots, K$. Here, $n_i(t^{\rho_i(n)})$ counts the completed prediction–evaluation steps at task initialization, before any observation in the new task is processed. Thus, t indexes local observations, whereas $t^{\rho_i(n)}$ denotes system time. For the first observation with $k = 1$, the local model and threshold are initialized as $\boldsymbol{\theta}_t^i := \boldsymbol{\theta}^{\rho_i(n)},\ q_t^i := q^{\rho_i(n)}$.

At each local step, client i observes $\boldsymbol{x}_t^i$ and constructs a prediction set using its current model $\boldsymbol{\theta}_t^i$ and calibration threshold q_t^i: $C_t^i := \left\{y \in \mathcal{Y} : s_t^i(\boldsymbol{x}_t^i, y) \le q_t^i\right\}$ with $C_t^i = \varnothing$ when $q_t^i < 0$. Once the response is observed, the client computes the realized score and miscoverage indicator that defined $S_t^i := s_t^i(\boldsymbol{x}_t^i, y_t^i),\ \mathrm{err}_t^i := \mathbf{1}\{y_t^i \notin C_t^i\} = \mathbf{1}\{S_t^i > q_t^i\}$. The client then updates its model by SGD and its threshold by a subgradient step on the pinball loss as follows:

$$\begin{aligned} \boldsymbol{\theta}_{t+1}^i &= \boldsymbol{\theta}_t^i - \zeta_t \nabla_{\boldsymbol{\theta}} \ell\big(\boldsymbol{\theta}_t^i; \boldsymbol{x}_t^i, y_t^i\big)\,, \\ q_{t+1}^i &= q_t^i + \eta_t \left(\mathrm{err}_t^i - \alpha\right), \end{aligned} \tag{2}$$

where $\zeta_t, \eta_t > 0$ are the step sizes indexed by the client's observation count.

For $k < K$, the client uses the updated model and threshold in Equation (2) for its next prediction. The client repeats this updating procedure until all K observations have been processed, without incorporating new global states during the task. After the final step, corresponding to $t = n_i(t^{\rho_i(n)}) + K$, client i computes its cumulative model and calibration increments relative to the initialization state:

$$\Delta_{\boldsymbol{\theta}_i}(\rho_i(n)) := \boldsymbol{\theta}_{t+1}^i - \boldsymbol{\theta}^{\rho_i(n)},\ \ \Delta_{q_i}(\rho_i(n)) := q_{t+1}^i - q^{\rho_i(n)}. \tag{3}$$

The client transmits its model and calibration increments to the server and waits for aggregation. At time t^{n+1}, the server aggregates the updates from $\mathcal{S}_n$ and returns the resulting global state $(\boldsymbol{\theta}^{n+1}, q^{n+1})$ to the participating clients. Upon receipt, client i initializes its next local task by replacing its local model and threshold with $\boldsymbol{\theta}_{t+1}^i := \boldsymbol{\theta}^{n+1},\ q_{t+1}^i := q^{n+1}$ before processing its next observation.

3.3 Server Aggregation Rule

To account for unequal participation frequencies under fixed-interval aggregation, we adopt the weighting rule in Fraboni et al. (2023): $d_i := \lceil \frac{\tau_i}{\Delta_t} \rceil p_i$, and $\omega_i(n) := d_i\, \mathbf{1}\{i \in \mathcal{S}_n\}$, where p_i is the prescribed client importance and $\omega_i(n)$ is the effective aggregation weight in round n. Under the fixed-duration timing model, client i contributes once every $\lceil \tau_i/\Delta_t \rceil$ server rounds. The multiplier compensates for less frequent participation by assigning larger weights to clients with longer participation periods. Specifically, let W be a common multiple of the participation periods $\{\lceil \tau_i/\Delta_t \rceil\}_{i=1}^M$. Over a complete cycle of W server rounds, $\sum_{n=0}^{W-1} \omega_i(n)/W = p_i,\ i = 1, \ldots, M$. Accordingly, each client's average aggregation weight equals its prescribed importance, although the weights need not sum to one in an individual round.

At time t^{n+1}, the server aggregates the model and calibration increments defined in Equation (3) using the client weights d_i:

$$\begin{aligned} \boldsymbol{\theta}^{n+1} &= \boldsymbol{\theta}^n + \zeta_{t^{n+1}} \sum_{i \in \mathcal{S}_n} d_i\, \Delta_{\boldsymbol{\theta}_i}(\rho_i(n))\,, \\ q^{n+1} &= q^n + \gamma_{t^{n+1}} \sum_{i \in \mathcal{S}_n} d_i\, \Delta_{q_i}(\rho_i(n))\,, \end{aligned}$$

where $\varsigma_{t^{n+1}} > 0$ and $\gamma_{t^{n+1}} > 0$ are the server step sizes for model and calibration updates, respectively. If $\mathcal{S}_n = \varnothing$, the global state remains unchanged. After aggregation, participating clients receive $(\boldsymbol{\theta}^{n+1}, q^{n+1})$ and initialize their next local tasks from this updated state, while nonparticipating clients continue their ongoing tasks. Appendix C presents detailed pseudocode for the asynchronous server-side aggregation procedure.

4 THEORETICAL ANALYSIS

In this section, we analyze the instantaneous conditional coverage and long-run empirical coverage of our proposed OAFCP. We first specify the information structure and define the coverage criteria. We then state the assumptions and establish the corresponding theoretical guarantees.

Let (i_ℓ, t_ℓ) denote the client and local observation indices associated with the ℓth response revelation. Define the global filtration by $\mathcal{F}_\ell := \mathcal{F}_0 \vee \sigma(i_j, t_j, \boldsymbol{x}_{t_j}^{i_j}, y_{t_j}^{i_j} : 1 \le j \le \ell),\ \ell \ge 1$, where $\mathcal{F}_0$ contains the initial state, fixed scheduling information, and independent algorithmic random seeds. Here, $\sigma(\cdot)$ denotes the generated σ-algebra, and $\vee$ denotes the smallest σ-algebra containing both arguments. We assume that the client and task indices, scheduled aggregation round, score function, and threshold associated with the ℓth revelation are $\mathcal{F}_{\ell-1}$-measurable. For $(i,t) = (i_\ell, t_\ell)$ corresponds to client i's tth observation, we write $\mathcal{F}_{t-1}^i := \mathcal{F}_{\ell-1}$. Recall the realized score $S_t^i := s_t^i(\boldsymbol{x}_t^i, y_t^i)$ and miscoverage indicator $\mathrm{err}_t^i = \mathbf{1}\{S_t^i > q_t^i\}$. For client $i \in \mathcal{S}_n$, we use the task-level notation:

$$S_{i,k}^{\rho_i(n)} := S_t^i,\ q_{i,k-1}^{\rho_i(n)} := q_t^i,\ \mathrm{err}_{i,k}^{\rho_i(n)} := \mathrm{err}_t^i,\ \mathcal{F}_{i,k-1}^{\rho_i(n)} := \mathcal{F}_{t-1}^i,\ \ t = n_i(t^{\rho_i(n)}) + k,\ k = 1, \dots, K.$$

We first define the empirical coverage criteria used in the long-run analysis. Let $\Lambda_N := K \sum_{n=0}^{N-1} \sum_{i \in \mathcal{S}_n} d_i$ denote the aggregation-weighted number of observations represented in the first N server aggregations. For $\Lambda_N > 0$, define

$$\mathrm{Cov}_d^{\mathrm{agg}}(N) := \frac{1}{\Lambda_N} \sum_{n=0}^{N-1} \sum_{i \in \mathcal{S}_n} d_i \sum_{k=1}^{K} \left(1 - \mathrm{err}_{i,k}^{\rho_i(n)}\right).$$

This criterion includes only predictions from tasks whose updates have been aggregated by time t^N. To include all completed prediction evaluation steps, let $N_d(T) := \sum_{i=1}^{M} d_i n_i(T)$ be their weighted count by system time T. For $N_d(T) > 0$, define

$$\mathrm{Cov}_d(T) := \frac{\sum_{i=1}^{M} d_i \sum_{t=1}^{n_i(T)} \mathbf{1}\{y_t^i \in C_t^i\}}{N_d(T)}.$$

Unlike $\mathrm{Cov}_d^{\mathrm{agg}}(N)$, this measure also includes completed predictions whose task updates have not yet been aggregated. Both criteria weight observations by d_i and are generally different from the unweighted pooled-coverage objective in Section 2.

We next define instantaneous coverage, which concerns individual prediction events. Let $F_t^i(u) := \mathbb{P}(S_t^i \le u | \mathcal{F}_{t-1}^i),\ u \in \mathbb{R}$, denote the conditional distribution function of the realized score. Since q_t^i is $\mathcal{F}_{t-1}^i$-measurable, the instantaneous conditional coverage satisfies

$$\mathrm{Coverage}_t^i := \mathbb{P}\big(y_t^i \in C_t^i \,\big|\, \mathcal{F}_{t-1}^i\big) = F_t^i(q_t^i) = \mathbb{E}\big[1 - \mathrm{err}_t^i \,\big|\, \mathcal{F}_{t-1}^i\big]. \tag{4}$$

Under the task-level indexing defined above, we write $F_{i,k}^{\rho_i(n)}(u) := F_t^i(u),\ \mathrm{Coverage}_{i,k}^{\rho_i(n)} := \mathrm{Coverage}_t^i$. To summarize instantaneous coverage across asynchronous clients, consider a horizon N at which every client has contributed at least once. Define $\nu_i(N) := \max\{n \in \{0, \dots, N-1\} : i \in \mathcal{S}_n\}$ as client i's most recent participating round. The corresponding task was initialized in round $\rho_i(\nu_i(N))$. Define the client-weighted instantaneous coverage as

$$\mathrm{Coverage}_{p,k}(N) := \sum_{i=1}^{M} p_i\ \mathrm{Coverage}_{i,k}^{\rho_i(\nu_i(N))}, \qquad k = 1, \dots, K.$$

This quantity averages the conditional coverage probabilities at the kth observation of each client's most recently aggregated task. We next state the assumptions used to analyze these coverage criteria.

Assumption 1. *There exists some positive constant B such that the nonconformity scores satisfy $0 \le S_t^i \le B$ for every client i and processed observation t with initial threshold $q^0 \in [0, B]$.*

Assumption 2. *There exists a fixed integer $\tau < \infty$ such that $0 \le n - \rho_i(n) \le \tau,\ n \ge 0,\ i \in \mathcal{S}_n$.*

Assumption 3. *For constants $\eta_0, \gamma_0 > 0$, the local and server calibration step sizes satisfy $\eta_t = \eta_0(1+t)^{-a},\ \gamma_{t^{n+1}} = \gamma_0(n+1)^{-b}$, where $0 < a < 1,\ b \ge 0,\ 1/2 < a + b < 1$. Specifically, $\eta_{i,k}^{\rho_i(n)} = \eta_{\rho_i(n)},\ i \in \mathcal{S}_n,\ k = 1, \dots, K$.*

Assumption 4. *There exist constants $\kappa > 0$ and an integer $N_0 \geq 1$ such that $\Lambda_N := K \sum_{n=0}^{N-1} \sum_{i \in \mathcal{S}_n} d_i \geq \kappa N,\ N \geq N_0$.*

Assumption 1 imposes a uniform bound on the nonconformity scores while allowing them to vary across clients and over time; see Angelopoulos et al. (2024). Assumption 2 places a uniform bound on update staleness, paralleling the finite-answering-time condition of Fraboni et al. (2023). Assumption 3 specifies polynomially decaying local and server calibration step sizes; see (Angelopoulos et al., 2024) and effective step-size conditions in local SGD (Li et al., 2022). Under the fixed-duration schedule, each client's observation count grows linearly with the server-round index, so the effective calibration step size scales as $n^{-(a+b)}$. The condition $1/2 < a + b < 1$ ensures that this sequence is nonsummable while its squared sequence is summable. Assumption 4 requires the cumulative aggregation-weighted sample count to grow at least linearly with the number of server rounds, ensuring sufficient calibration feedback at the system level.

Theorem 1. *Suppose Assumptions 1–4 hold. Then, for every $N \geq 1$ such that $\Lambda_N > 0$,*

$$|\mathrm{Cov}_d^{\mathrm{agg}}(N) - (1-\alpha)| \leq (C_q \lambda_{N-1}^{-1} + R_N)/\Lambda_N. \tag{5}$$

Moreover, for system time T with $N = \lfloor T/\Delta t \rfloor$ and $N_d(T) > 0$,

$$|\mathrm{Cov}_d(T) - (1-\alpha)| \leq (C_q \lambda_{N-1}^{-1} + R_N + KD)/N_d(T), \tag{6}$$

where the effective calibration step size is $\lambda_n = \gamma_0 \eta_0 (n+1)^{-b}(1 + n\Delta t + K)^{-a},\ n \geq 0$, $D = \sum_{i=1}^{M} d_i,\ A = \gamma_0 K \eta_0 D,\ H = \tau A + (K-1)\eta_0,\ C_q = B + 2H + A$, and $\mathcal{R}_N = \sum_{n=0}^{N-1} \sum_{i \in S_n} d_i \sum_{k=1}^{K} (\eta_{t^{\rho_i(n)}+k}/\eta_{t^n+K} - 1)$.

Theorem 1 establishes the long-run coverage guarantee in the asynchronous federated setting. In the special case $M = K = 1$, with unit aggregation weight, full participation, and no communication delay, the result reduces to the standard online conformal coverage guarantee.

Assumption 5. *There exist deterministic distribution functions $F^1, \ldots, F^M$ such that for every client i and continuity point u of F_i, $\max_{1 \leq k \leq K} \left| F_t^i(u) - F^i(u) \right| \longrightarrow 0$ as $n \to \infty$ along the server rounds satisfying $i \in \mathcal{S}_n$.*

Assumption 6. *For $\alpha \in (0,1)$, define the client-weighted limiting score distribution and its $(1-\alpha)$-quantile by $F(u) := \sum_{i=1}^{M} p_i F^i(u),\ q^* := \inf\{u \in \mathbb{R} : F(u) \geq 1-\alpha\}$. Assume that, for every $\varepsilon > 0$, $F(q^* - \varepsilon) < 1 - \alpha < F(q^* + \varepsilon)$.*

Assumption 5 formulates score-distribution stability in terms of the conditional distributions associated with asynchronous client predictions. It extends the stability perspective of Angelopoulos et al. (2024) while allowing heterogeneous client-specific limits. Assumption 6 identifies a unique target quantile of the limiting mixture.

Theorem 2. *Suppose Assumptions 1, 2, 4, and Assumptions 5–6 hold. Then the global calibration threshold generated by OAFCP satisfies $q^n \longrightarrow q^*$, almost surely as $n \to \infty$. Moreover, let $q^i(T)$ denote the threshold stored by client i at system time T. Then $\max_{1 \leq i \leq M} \left| q^i(T) - q^* \right| \longrightarrow 0$, almost surely as $T \to \infty$.*

Theorem 2 shows convergence of the global and local thresholds to a common limit. Next, we establish the instantaneous coverage based on Theorem 2

Corollary 1. *Suppose the conditions of Theorem 2 hold and each F^i is continuous at q^*. Then, as $n \to \infty$,*

$$\max_{\substack{i \in \mathcal{S}_n \\ 1 \leq k \leq K}} \left| \mathrm{Coverage}_t^i - F^i(q^*) \right| \longrightarrow 0, \ \textit{almost surely}, \tag{7}$$

where the maximum is defined as zero when $\mathcal{S}_n = \varnothing$. Furthermore, as $N \to \infty$,

$$\max_{1 \leq k \leq K} \left| \mathrm{Coverage}_{p,k}(N) - (1-\alpha) \right| \longrightarrow 0, \ \textit{almost surely}. \tag{8}$$

In summary, Theorem 2 and Corollary 1 establish asymptotic calibration of OAFCP under asynchronous federated updates. Despite delayed communication and heterogeneous client states, each client's instantaneous coverage converges to $F^i(q^*)$, while the client-weighted coverage converges to the nominal level $1 - \alpha$, provided that the client-specific score distributions stabilize over time.

5 EXPERIMENT

In this section, we evaluate the finite-sample performance of the proposed method through simulation studies and six real-data analysis. We compare the proposed OAFCP with online synchronous federated conformal prediction (OSFCP), which waits for all clients to complete one local update before aggregation. Under OSFCP, clients incur no latency, and the aggregation weights equal the importance weights. We set the significance level to $\alpha = 0.1$. Details of the data-generating process are provided in Appendix A.

The server aggregates every $\Delta_t = 2$ time units, with $K = 10$ and $M = 10$. Each client has a processing speed τ_i. We use FX to represent different levels of hardware heterogeneity, where FX indicates that the fastest client is $X\%$ faster than the slowest.

Example 1 (**Regression**). We consider a multiple linear regression model with feature dimension $p = 4$ under three settings:

- **Setting 1-Constant coefficients**: The coefficient vector remains constant over time.
- **Setting 2-Abrupt changes**: The coefficient vector is piecewise constant with two changes at prespecified time points.
- **Setting 3-Gradual drift**: The coefficient vector changes smoothly through linear interpolation.

Figure 3 shows the long-run coverage, rolling coverage, loss, and interval width hardware heterogeneity of $F90$. The relevant results regarding the hardware heterogeneity of $F30$ and $F70$ are provided in the Appendix E. Both methods attain long-run coverage close to the target level $1-\alpha = 0.9$. Under constant coefficients and gradual drift, their loss and interval width are also comparable. Under abrupt changes, however, OAFCP adapts more rapidly, with rolling coverage recovering faster and exhibiting smaller fluctuations, together with lower loss and narrower intervals. Compared with F30, the advantage of OAFCP in interval width becomes more pronounced under F90. Because greater hardware heterogeneity increases the synchronization delay in OSFCP, whereas OAFCP can incorporate available client updates asynchronously and update the quantile more promptly.

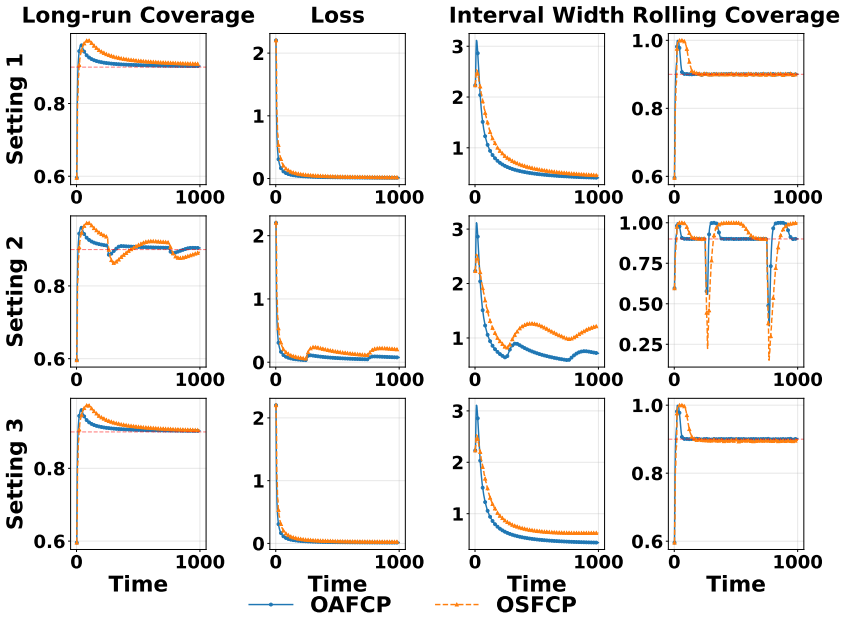


Figure 3: Regression simulation results under F90.

Example 2 (**Classification**). Three settings are considered in the calssification task.

- **Setting 1-Constant coefficients**: The coefficient vector remains constant over time.
- **Setting 2-Gradual drift**: The coefficient vector changes smoothly through linear interpolation.
- **Setting 3-Class emergence**: A new class that gradually emerges by activating a specific feature,while the other classes remain fixed.

The aggregation mechanism remains the same to that in Example 1. Similar patterns are observed for the classification task, with OAFCP and OSFCP showing comparable performance under the constant-coefficient and gradual-drift settings. Under class emergence, OAFCP yields slightly higher loss and wider prediction sets, reflecting a more conservative response to the newly emerging class. However, as client-side hardware heterogeneity has increased, this interval conservatism has diminished. Detailed classification results are provided in Appendix E. Figure 4 further presents the aggregated quantile and instantaneous coverage under the constant-coefficient setting for both regression and classification. In both tasks, the aggregated quantile stabilizes rapidly, while the instantaneous coverage remains close to the target level.

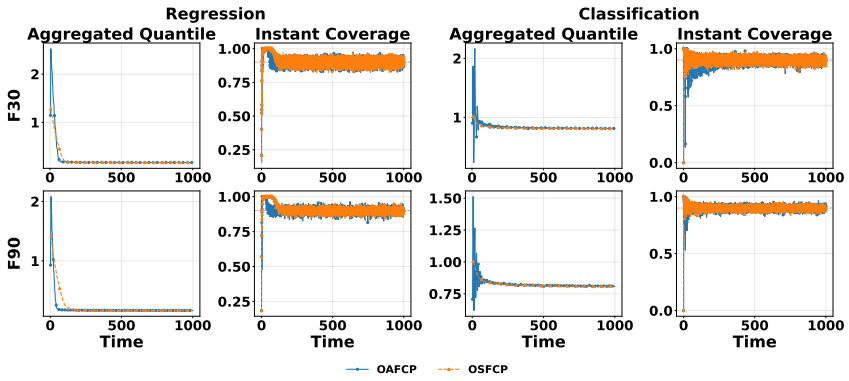


Figure 4: Aggregation quantile and instant coverage in the constant coefficients setting.

Example 3 (**Real-data based simulation**). We evaluate OAFCP on six publicly available datasets covering both regression and classification tasks: **DJIA Stocks** (Szrlee, 2018), **Beijing Air** (Chen, 2017), **WESAD** (Schmidt et al., 2018), **MNIST** (LeCun et al., 1998), **Fashion-MNIST** (Xiao et al., 2017), and **CIFAR-10** (Krizhevsky, 2009). The first three datasets admit natural client-level partitions. For the remaining three, we distribute samples across 10 clients according to a Dirichlet(0.1) distribution to induce heterogeneous, non-IID client data. Detailed descriptions of the real-world datasets are provided in Appendix B.

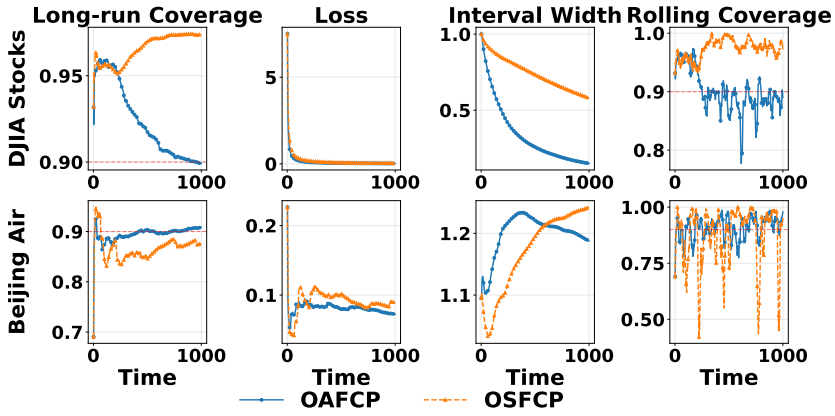


Figure 5: Results on real regression datasets under F90.

Figure 5 reports the regression results under $F90$, while the classification and the other regression results are provided in Appendix E. OAFCP maintains long-run coverage close to the target level across the datasets and generally exhibits more stable rolling coverage. For regression, OSFCP shows noticeable overcoverage on DJIA Stocks and undercoverage on Beijing Air, whereas OAFCP remains closer to the nominal level and yields substantially narrower intervals on DJIA Stocks while remaining comparable on Beijing Air. For classification, both methods perform similarly on MNIST and Fashion-MNIST. OAFCP achieves lower loss on WESAD, while on CIFAR-10 it produces slightly wider prediction sets and higher loss, accompanied by coverage closer to the target level.

AI USE STATEMENT

The authors used a general-purpose LLM assistant for language editing of author-written prose (grammar, word choice, and conciseness). The research questions, theoretical developments, study design, data collection, statistical analyses, and interpretation were conceived and conducted entirely by the authors. All model-generated suggestions were critically reviewed, verified, and revised by the authors, who assume full responsibility for the content of the manuscript, including any remaining errors.

REPRODUCIBILITY STATEMENT

We are committed to ensuring the reproducibility of our results. The simulation settings and the related data descriptions are provided in the appendix. The real-world datasets used in our work are publicly available and properly cited. We report the detailed parameters for the different simulation scenario settings in Appendix A. Appendix B provides a brief introduction to the six real-world datasets used in this paper. All experiments were conducted in a Python 3.11 environment, running on a Windows 10 Enterprise 22H2 machine equipped with an 11th Gen Intel(R) Core(TM) i5-1135G7 CPU (2.40 GHz). Parallel computations were performed on the CPU using Python's ThreadPoolExecutor. A random seed of 2024 was used for the simulation experiments, with statistics averaged over 200 repeated runs. For all real-world datasets, a seed of 2024 was used. The detailed proofs of the theoretical results are presented in Appendix D. Due to space limitations, all experimental supplementary result tables and plots of this work are provided in Appendix E.

REFERENCES

Anastasios N. Angelopoulos, Emmanuel J. Candes, and Ryan J. Tibshirani. Conformal PID control for time series prediction. In *Advances in Neural Information Processing Systems*, volume 36, pp. 23047–23074. Curran Associates, Inc., 2023.

Anastasios Nikolas Angelopoulos, Rina Barber, and Stephen Bates. Online Conformal Prediction with Decaying Step Sizes. In *International Conference on Machine Learning*, volume 235, pp. 1616–1630, 2024. URL https://mlanthology.org/icml/2024/angelopoulos2024icml-online/.

Yajie Bao, Michael Crawshaw, Shan Luo, and Mingrui Liu. Fast composite optimization and statistical recovery in federated learning. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 1508–1536. PMLR, 2022.

Beepul Bharti, Ambar Pal, Jacopo Teneggi, and Jeremias Sulam. Parameter-free and group conditional online conformal prediction, 2026. URL https://arxiv.org/abs/2606.00419.

Aadyot Bhatnagar, Huan Wang, Caiming Xiong, and Yu Bai. Improved online conformal prediction via strongly adaptive online learning. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 2337–2363. PMLR, 2023.

Song Chen. Beijing Multi-Site Air Quality. UCI Machine Learning Repository, 2017. DOI: https://doi.org/10.24432/C5RK5G.

Yann Fraboni, Richard Vidal, Laetitia Kameni, and Marco Lorenzi. A general theory for federated optimization with asynchronous and heterogeneous clients updates. *Journal of Machine Learning Research*, 24(110):1–43, 2023.

François Gauthier, Vinay Chakravarthi Gogineni, Stefan Werner, Yih-Fang Huang, and Anthony Kuh. Asynchronous online federated learning with reduced communication requirements. *IEEE Internet of Things Journal*, 10(23):20761–20775, 2023. doi: 10.1109/JIOT.2023.3314923.

Pouya M. Ghari and Yanning Shen. Personalized online federated learning with multiple kernels. In *Advances in Neural Information Processing Systems*, volume 35, pp. 33316–33329, 2022.

Isaac Gibbs and Emmanuel Candès. Adaptive conformal inference under distribution shift. In *Advances in Neural Information Processing Systems*, volume 34, pp. 1660–1672. Curran Associates, Inc., 2021.

Eduard Gorbunov, Filip Hanzely, and Peter Richtárik. Local SGD: Unified theory and new efficient methods. In *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pp. 3556–3564. PMLR, 2021.

Songnam Hong and Jeongmin Chae. Communication-efficient randomized algorithm for multi-kernel online federated learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(12):9872–9886, 2022. doi: 10.1109/TPAMI.2021.3129809.

Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. Scaffold: Stochastic controlled averaging for federated learning. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 5132–5143. PMLR, 2020.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, Computer Science Department, University of Toronto, 2009. URL `https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf`.

Louis Leconte, Matthieu Jonckheere, Sergey Samsonov, and Eric Moulines. Queuing dynamics of asynchronous federated learning. In *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pp. 1711–1719. PMLR, 2024.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. doi: 10.1109/5.726791.

Tian Li, Anit Kumar Sahu, Ameet Talwalkar, and Virginia Smith. Federated learning: Challenges, methods, and future directions. *IEEE Signal Processing Magazine*, 37(3):50–60, 2020a. doi: 10.1109/MSP.2020.2975749.

Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. In *Proceedings of Machine Learning and Systems 2020*, 2020b.

Xiang Li, Jiadong Liang, Xiangyu Chang, and Zhihua Zhang. Statistical estimation and online inference via local SGD. In *Proceedings of Thirty Fifth Conference on Learning Theory*, volume 178 of *Proceedings of Machine Learning Research*, pp. 1613–1661. PMLR, 2022.

Yue Li, Sudip Bhujel, Cameron Lira, Ning Wang, and Yang Xiao. Model-consistent byzantine-resilient decentralized federated learning for collaborative missions, 2026. URL `https://arxiv.org/abs/2608.22552`.

Tuo Liu, Edgar Dobriban, and Francesco Orabona. Online conformal prediction via universal portfolio algorithms, 2026. URL `https://arxiv.org/abs/2602.03168`.

Othmane Marfoq, Giovanni Neglia, Laetitia Kameni, and Richard Vidal. Federated learning for data streams. In *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*, volume 206 of *Proceedings of Machine Learning Research*, pp. 8889–8924. PMLR, 2023.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pp. 1273–1282. PMLR, 2017.

John Nguyen, Kshitiz Malik, Hongyuan Zhan, Ashkan Yousefpour, Mike Rabbat, Mani Malek, and Dzmitry Huba. Federated learning with buffered asynchronous aggregation. In *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pp. 3581–3607. PMLR, 2022.

Philip Schmidt, Attila Reiss, Robert Duerichen, Claus Marberger, and Kristof Van Laerhoven. Introducing WESAD, a multimodal dataset for wearable stress and affect detection. In *Proceedings of the 20th ACM International Conference on Multimodal Interaction*, ICMI '18, pp. 400–408, New York, NY, USA, 2018. Association for Computing Machinery. doi: 10.1145/3242969.3242985.

Sebastian U. Stich. Local SGD converges fast and communicates little. In *International Conference on Learning Representations*, 2019.

Szrlee. Djia 30 stock time series. Kaggle, 2018. URL https://www.kaggle.com/datasets/szrlee/stock-time-series-20050101-to-20171231.

Vladimir Vovk, Alexander Gammerman, and Glenn Shafer. *Algorithmic Learning in a Random World*. Springer, New York, 2005. ISBN 978-0-387-00152-4.

Yujia Wang, Shiqiang Wang, Songtao Lu, and Jinghui Chen. Fadas: Towards federated adaptive asynchronous optimization. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 51701–51733. PMLR, 2024.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017. URL https://arxiv.org/abs/1708.07747.

Cong Xie, Oluwasanmi Koyejo, and Indranil Gupta. Asynchronous federated optimization, 2019. URL https://arxiv.org/abs/1903.03934.

Wenjing Yan, Xiangyu Zhong, Xiaolu Wang, and Ying-Jun Angela Zhang. Momentum-driven adaptivity: Towards tuning-free asynchronous federated learning. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 70410–70441. PMLR, 2025.

Qiangqiang Zhang, Chenfei Gu, Xinwei Feng, Jinhan Xie, and Ting Li. Online locally differentially private conformal prediction via binary inquiries. In *Advances in Neural Information Processing Systems*, 2025a.

Qiangqiang Zhang, Ting Li, Xinwei Feng, Xiaodong Yan, and Jinhan Xie. Online differentially private conformal prediction for uncertainty quantification. In *Proceedings of the 42nd International Conference on Machine Learning*. JMLR.org, 2025b.

Zhiyu Zhang, Zhou Lu, and Heng Yang. The benefit of being bayesian in online conformal prediction. In *NeurIPS 2024 Workshop on Bayesian Decision-making and Uncertainty*, 2024.

A Simulation Setting

Sections A.1–A.2 provide the detailed synthetic-data generation procedures for the simulation studies in Examples 1 and 2 of Section 5. The local learning rate at observation t is $\eta_t = \eta_0/(t+1)^{0.505}$, with $\eta_0 = 0.5$ for linear regression and $\eta_0 = 2$ for classification. The server learning rate is fixed throughout the experiments at $\gamma_t = \gamma_0 = 1$.

A.1 Regression Simulation

Example (Regression). We consider three settings for the time-varying coefficient vector used in the simulations. In all settings, the feature dimension is fixed at $p = 4$, and the intercept is set to $\beta_0 = 1$. We write $\boldsymbol{\beta}(t) := [\beta_0, \beta_1(t), \beta_2(t), \beta_3(t), \beta_4(t)]^\top$ for the time-varying coefficient vector.

- **Setting 1-Stationary Distribution**: The coefficient vector remains constant over time:
$$\boldsymbol{\beta}(t) = [1,\ 2,\ 1,\ 0,\ 0]^\top, \quad \forall t \in [0, T]$$
This setting represents a stationary environment in which the underlying data-generating mechanism does not change over time.

- **Setting 2-Abrupt Changes**: The coefficient vector is piecewise constant and undergoes two abrupt changes at predefined changepoints:
$$\boldsymbol{\beta}(t) = \begin{cases} [1,\ 2,\ 1,\ 0,\ 0]^\top, & 0 \le t \le t_1 \\ [1,\ 0,\ -2,\ -1,\ 0]^\top, & t_1 < t \le t_2 \\ [1,\ 0,\ 0,\ 2,\ 1]^\top, & t_2 < t \le T, \end{cases}$$
where $t_1 = T/4$ and $t_2 = 3T/4$. This setting models abrupt concept drift, with the underlying data-generating mechanism changing discontinuously at the two changepoints.

- **Setting 3-Gradual Drift**: The coefficient vector evolves smoothly from an initial state to a final state through linear interpolation:
$$\boldsymbol{\beta}(t) = \left(1 - \frac{t}{T}\right)\boldsymbol{\beta}_{\text{start}} + \frac{t}{T}\boldsymbol{\beta}_{\text{end}}, \quad t \in [0, T],$$
where
$$\boldsymbol{\beta}_{\text{start}} = [1,\ 2,\ 1,\ 0,\ 0]^\top, \qquad \boldsymbol{\beta}_{\text{end}} = [1,\ 0,\ 0,\ 2,\ 1]^\top.$$
This setting models gradual concept drift, with the underlying data-generating mechanism changing continuously over time.

Client Data Distribution. To simulate heterogeneous, non-IID data across clients, we assign each client a distinct feature distribution: $\boldsymbol{x}^i \sim \mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$ where $\boldsymbol{\mu}_i \in \mathbb{R}^4$ and $\boldsymbol{\Sigma}_i \in \mathbb{R}^{4\times 4}$ are the client-specific mean vector and covariance matrix, respectively. We generate
$$\boldsymbol{\mu}_i \sim \mathcal{N}(\mathbf{0}, 0.5^2\mathbf{I}_4), \quad \boldsymbol{\Sigma}_i = \frac{\mathbf{A}_i\mathbf{A}_i^\top}{\max|\mathbf{A}_i\mathbf{A}_i^\top|} \times 0.5,$$
where $\boldsymbol{A}_i \in \mathbb{R}^{4\times 4}$ has i.i.d. $\mathcal{N}(0,1)$ entries. For client i at time t, the response is generated according to
$$y_t^i = [1, \boldsymbol{x}_t^{i\top}]\beta(t) + \varepsilon_t^i, \quad \varepsilon_t^i \sim \mathcal{N}(0, 0.1^2).$$
Thus, client heterogeneity is induced through client-specific feature distributions, while temporal distribution shift is controlled by the time-varying coefficient vector $\boldsymbol{\beta}(t)$.

A.2 Classification Simulation

Example (Classification). We also consider three settings for the classification task. In all cases, the feature dimension is fixed at $p = 3$, and no intercept term is included. The response variable has three classes, $y_t^i \in \{0, 1, 2\}$. For each class $c \in \{0, 1, 2\}$, let $\boldsymbol{\beta}^c(t) := [\beta_1^c(t), \beta_2^c(t), \beta_3^c(t)]^\top$ denote the corresponding time-varying coefficient vector.

- **Setting 1-Stationary Distribution**: In the stationary setting, the class-specific coefficient vectors remain constant over time:

$$\boldsymbol{\beta}^0(t) = [-1, 0, 0]^\top, \quad \boldsymbol{\beta}^1(t) = [1, 0, 0]^\top, \quad \boldsymbol{\beta}^2(t) = [0, 0, 1]^\top, \quad \forall t \in [0, T].$$

- **Setting 2-Gradual Drift**: In the gradual-drift setting, each class-specific coefficient vector evolves smoothly from its initial state to its final state through linear interpolation. For each class $c \in \{0, 1, 2\}$, the coefficient vector evolves from an initial state to a final state through linear interpolation:

$$\boldsymbol{\beta}^c(t) = \left(1 - \frac{t}{T}\right) \boldsymbol{\beta}^c_{\text{start}} + \frac{t}{T} \boldsymbol{\beta}^c_{\text{end}}, \qquad t \in [0, T].$$

  The initial and final coefficient vectors are

$$\begin{aligned} \boldsymbol{\beta}^0_{\text{start}} &= [-2,\ 0,\ 0]^\top, & \boldsymbol{\beta}^0_{\text{end}} &= [2,\ 0,\ 0]^\top, \\ \boldsymbol{\beta}^1_{\text{start}} &= [2,\ 0,\ 0]^\top, & \boldsymbol{\beta}^1_{\text{end}} &= [-2,\ 0,\ 0]^\top, \\ \boldsymbol{\beta}^2_{\text{start}} &= [0,\ 0,\ 2]^\top, & \boldsymbol{\beta}^2_{\text{end}} &= [0,\ 0,\ -2]^\top. \end{aligned}$$

  This setting models gradual concept drift by continuously changing the class-specific decision structure over time.

- **Setting 3-Class Emergence**: This setting models the gradual emergence of a new class through the activation of a previously inactive feature, while the remaining classes stay fixed. We consider $C = 4$ classes and $p = 5$ features. The coefficient vectors for the first three classes are constant over time:

$$\begin{aligned} \boldsymbol{\beta}^0(t) &= [2,\ 0,\ 0,\ 0,\ 0]^\top, & \boldsymbol{\beta}^1(t) &= [-2,\ 0,\ 0,\ 0,\ 0]^\top, \\ \boldsymbol{\beta}^2(t) &= [0,\ 0,\ 2,\ 0,\ 0]^\top, & & \forall\, t \in [0, T]. \end{aligned}$$

  For the emerging class $c = 3$, the coefficient vector evolves from an inactive state to a feature-specific state:

$$\boldsymbol{\beta}^3_{\text{start}} = [0,\ 0,\ 0,\ 0,\ 0]^\top, \qquad \boldsymbol{\beta}^3_{\text{end}} = [0,\ 0,\ 0,\ 0,\ 4]^\top,$$

  with

$$\boldsymbol{\beta}^3(t) = \left(1 - \frac{t}{T}\right) \boldsymbol{\beta}^3_{\text{start}} + \frac{t}{T} \boldsymbol{\beta}^3_{\text{end}}, \qquad t \in [0, T].$$

  Thus, the fourth class becomes progressively distinguishable over time through the increasing influence of the fifth feature.

Client Data Distribution For the classification experiments, we use the same client-specific feature-generation mechanism as in the regression setting. Each client is assigned a distinct feature distribution, thereby inducing non-IID covariates across clients.

B REAL DATASET INTRODUCTION

In this appendix, we briefly introduce the six real-world datasets used in Example 3 of Section 5.

DJIA Stocks. Daily trading records for 31 constituent companies of the Dow Jones Industrial Average span January 2006 to January 2018, yielding approximately 93,612 observations. Each stock is treated as an individual client, and the regression task is to predict its next-day closing price from the available historical features; see Szrlee (2018).

Beijing Air Quality. Hourly measurements of six air pollutants and six meteorological variables were collected from 12 monitoring stations across Beijing between March 2013 and February 2017, resulting in 420,768 records. Each monitoring station is treated as a client, and $PM_{2.5}$ concentration is used as the regression target; see Chen (2017).

WESAD. Multimodal physiological signals were collected from multiple subjects under different affective conditions using wearable sensors. The natural subject-level organization induces heterogeneous client distributions because physiological responses vary substantially across individuals.

We treat each subject as a separate client and perform classification using signals collected from the chest-worn sensors; see Schmidt et al. (2018).

MNIST. The data consist of 70,000 grayscale handwritten-digit images of size 28×28, divided into 10 digit classes. Each image is flattened into a 784-dimensional feature vector and normalized to $[0, 1]$. To induce non-IID client distributions, samples are partitioned across clients according to a Dirichlet distribution with concentration parameter 0.1; see LeCun et al. (1998).

Fashion-MNIST. The data consist of 70,000 grayscale images of size 28×28 across 10 clothing categories. We apply the same flattening and normalization procedure as for MNIST and partition samples across clients using the same Dirichlet-based scheme; see Xiao et al. (2017).

CIFAR-10. The data consist of 60,000 color images of size $32 \times 32 \times 3$ across 10 object classes. To make the data compatible with our linear models, each image is converted to grayscale, flattened into a 1,024-dimensional feature vector, and normalized to $[0, 1]$. Samples are then partitioned across clients using the same Dirichlet-based procedure as for MNIST and Fashion-MNIST; see Krizhevsky (2009).

C Asynchronous Server Aggregation Algorithm

This appendix provides detailed pseudocode for the asynchronous server-side aggregation procedure underlying OAFCP.

Algorithm 1 Asynchronous Server Aggregation in OAFCP

Require: Number of clients M; aggregation interval Δ_t; task and delivery durations $\{\tau_i\}_{i=1}^M$; client importance weights $\{p_i\}_{i=1}^M$. Initial local step sizes ζ_0, η_0; initial server step sizes ζ_0 for model updates and γ_0 for calibration updates. Target coverage level $1 - \alpha$; initial global model $\boldsymbol{\theta}^0$ and calibration threshold q^0.

for $n = 0, \dots, N$ **do**

Clients in $\mathcal{S}_n = \{i : T_i^n \le \Delta_t\}$ transmit their completed local updates:

- $\Delta_{\boldsymbol{\theta}_i}(\rho_i(n)) = \boldsymbol{\theta}_{t+1}^i - \boldsymbol{\theta}^{\rho_i(n)}$.
- $\Delta_{q_i}(\rho_i(n)) = q_{t+1}^i - q^{\rho_i(n)}$.

At time t^{n+1}, the server updates the global model and calibration threshold:

- $\boldsymbol{\theta}^{n+1} = \boldsymbol{\theta}^n + \zeta_{t^{n+1}} \sum_{i \in \mathcal{S}_n} d_i \, \Delta_{\boldsymbol{\theta}_i}(\rho_i(n))$.
- $q^{n+1} = q^n + \gamma_{t^{n+1}} \sum_{i \in \mathcal{S}_n} d_i \, \Delta_{q_i}(\rho_i(n))$.

The server returns $(\boldsymbol{\theta}^{n+1}, q^{n+1})$ to clients in $\mathcal{S}_n$ to initialize their next local tasks.

end for

D Proofs of Theoretical Results

This appendix provides the detailed proofs of the theorems and corollaries stated in Section 4. In the proof, we employ the task-level notation

D.1 Proof of Lemma 1

Lemma 1. *Suppose that the nonconformity scores are uniformly bounded, i.e.,* $0 \le S_t^i \le B$*, for all* $i \in S_n$*,* $k = 1, \dots, K$*, and* $n \ge 0$*. Assume that* $q^0 \in [0, B]$*, and let* $\{\eta_t\}_{t \ge 0}$ *and* $\{\gamma_{t^{n+1}}\}_{n \ge 0}$ *be nonnegative sequences. Then, for all* $n \ge 0$*,*

$$-H - \alpha A \le q^n \le B + H + (1 - \alpha)A. \tag{9}$$

Consequently, for any $r, s \ge 0$*,*

$$|q^r - q^s| \le B + 2H + A =: C_q. \tag{10}$$

Proof of Lemma 1. For client $i \in \mathcal{S}_n$, the kth observation of the task initialized at $t^{\rho_i(n)}$ has local index $t = n_i(t^{\rho_i(n)}) + k$. With initialization $q_{i,0}^{\rho_i(n)} = q^{\rho_i(n)}$, the local calibration update is

$$q_{i,k}^{\rho_i(n)} = q_{i,k-1}^{\rho_i(n)} + \eta_{n_i(t^{\rho_i(n)})+k} \left(\mathrm{err}_{i,k}^{\rho_i(n)} - \alpha \right), \qquad k = 1, \ldots, K.$$

Summing over the K local steps yields

$$q_{i,K}^{\rho_i(n)} - q^{\rho_i(n)} = \sum_{k=1}^{K} \eta_{n_i(t^{\rho_i(n)})+k} \left(\mathrm{err}_{i,k}^{\rho_i(n)} - \alpha \right).$$

Since $-\alpha \le \mathrm{err}_{i,k}^{\rho_i(n)} - \alpha \le 1 - \alpha$ and $0 \le \eta_t \le \eta_0$, we then obtain

$$-\alpha K \eta_0 \le q_{i,K}^{\rho_i(n)} - q^{\rho_i(n)} \le (1-\alpha) K \eta_0. \tag{11}$$

The server updates the global threshold according to

$$q^{n+1} = q^n + \gamma_{t^{n+1}} \sum_{i \in \mathcal{S}_n} d_i \left(q_{i,K}^{\rho_i(n)} - q^{\rho_i(n)} \right). \tag{12}$$

Combining Equation (11) with $0 \le \gamma_{t^{n+1}} \le \gamma_0$ and $\sum_{i \in \mathcal{S}_n} d_i \le D$ yields

$$-\alpha A \le q^{n+1} - q^n \le (1-\alpha) A. \tag{13}$$

In particular, we obtain $|q^{n+1} - q^n| \le A$.

We next control the discrepancy between the current global threshold and the stale initialization used by each participating client. By Assumption 2,

$$\left| q^n - q^{\rho_i(n)} \right| \le \sum_{m=\rho_i(n)}^{n-1} \left| q^{m+1} - q^m \right| \le (n - \rho_i(n))\, A \le \tau A.$$

Before processing its kth observation, client i has performed $k-1$ local calibration updates. Therefore, we have

$$\begin{aligned} \left| q_{i,k-1}^{\rho_i(n)} - q^{\rho_i(n)} \right| &\le \sum_{\ell=1}^{k-1} \eta_{n_i(t^{\rho_i(n)})+\ell} \left| \mathrm{err}_{i,\ell}^{\rho_i(n)} - \alpha \right| \\ &\le (k-1)\eta_0 \le (K-1)\eta_0. \end{aligned}$$

The triangle inequality then yields

$$\begin{aligned} \left| q_{i,k-1}^{\rho_i(n)} - q^n \right| &\le \left| q_{i,k-1}^{\rho_i(n)} - q^{\rho_i(n)} \right| + \left| q^{\rho_i(n)} - q^n \right| \\ &\le (K-1)\eta_0 + \tau A = H \end{aligned} \tag{14}$$

for every $i \in \mathcal{S}_n$ and $k = 1, \ldots, K$.

This bound determines the direction of the global update outside $[-H, B+H]$. If $q^n > B + H$, then Equation (14) implies $q_{i,k-1}^{\rho_i(n)} \ge q^n - H > B$ for every participating client and local step. Since $S_{i,k}^{\rho_i(n)} \le B$, all corresponding miscoverage indicators equal zero. Substituting into Equation (12), we obtain

$$q^{n+1} - q^n = -\alpha \gamma_{t^{n+1}} \sum_{i \in \mathcal{S}_n} d_i \sum_{k=1}^{K} \eta_{n_i(t^{\rho_i(n)})+k} \le 0. \tag{15}$$

Similarly, if $q^n < -H$, then $q_{i,k-1}^{\rho_i(n)} \le q^n + H < 0$. The nonnegativity of the scores implies that all corresponding miscoverage indicators equal one. Hence,

$$q^{n+1} - q^n = (1-\alpha) \gamma_{t^{n+1}} \sum_{i \in \mathcal{S}_n} d_i \sum_{k=1}^{K} \eta_{n_i(t^{\rho_i(n)})+k} \ge 0. \tag{16}$$

These inequalities also hold when $\mathcal{S}_n = \varnothing$, in which case the global threshold remains unchanged.

Set $L_q := -H - \alpha A$ and $U_q := B + H + (1-\alpha)A$. We prove by induction that $q^n \in [L_q, U_q]$ for all $n \geq 0$. The claim holds initially because $q^0 \in [0, B] \subseteq [L_q, U_q]$. Suppose it holds at round n. If $q^n \leq B + H$, then Equation (13) gives

$$q^{n+1} \leq q^n + (1-\alpha)A \leq B + H + (1-\alpha)A = U_q.$$

Otherwise, Equation (15) implies $q^{n+1} \leq q^n \leq U_q$.

For the lower bound, if $q^n \geq -H$, then Equation (13) yields

$$q^{n+1} \geq q^n - \alpha A \geq -H - \alpha A = L_q.$$

Otherwise, Equation (16) gives $q^{n+1} \geq q^n \geq L_q$. Thus, $q^{n+1} \in [L_q, U_q]$, we have completed the induction and proved Equation (9). Hence, since $q^r, q^s \in [L_q, U_q]$ for all nonnegative integers r, s,

$$|q^r - q^s| \leq U_q - L_q = B + 2H + A = C_q.$$

We have completed the proof of this lemma. □

D.2 Proof of Theorem 1

Proof of the Theorem 1. Fix an integer $N \geq 1$ such that $\Lambda_N > 0$. For each server round n, define the weighted miscoverage deviation

$$E_n := \sum_{i \in \mathcal{S}_n} d_i \sum_{k=1}^{K} \left(\mathrm{err}_{i,k}^{\rho_i(n)} - \alpha \right).$$

By the definitions of Λ_N and $\mathrm{Cov}_d^{\mathrm{agg}}(N)$,

$$\begin{aligned} \mathrm{Cov}_d^{\mathrm{agg}}(N) - (1-\alpha) &= \frac{1}{\Lambda_N} \sum_{n=0}^{N-1} \sum_{i \in \mathcal{S}_n} d_i \sum_{k=1}^{K} \left[1 - \mathrm{err}_{i,k}^{\rho_i(n)} - (1-\alpha) \right] \\ &= -\frac{1}{\Lambda_N} \sum_{n=0}^{N-1} E_n. \end{aligned} \tag{17}$$

It therefore suffices to bound the cumulative deviation $\sum_{n=0}^{N-1} E_n$.

For client $i \in \mathcal{S}_n$, the kth observation of the task initialized at $t^{\rho_i(n)}$ has local index $t = n_i(t^{\rho_i(n)}) + k$. Telescoping the local calibration updates in Equation (2) gives

$$q_{i,K}^{\rho_i(n)} - q^{\rho_i(n)} = \sum_{k=1}^{K} \eta_{n_i(t^{\rho_i(n)})+k} \left(\mathrm{err}_{i,k}^{\rho_i(n)} - \alpha \right).$$

Substituting this identity into the server update Equation (12), we obtain

$$q^{n+1} - q^n = \gamma_{t^{n+1}} \sum_{i \in \mathcal{S}_n} d_i \sum_{k=1}^{K} \eta_{n_i(t^{\rho_i(n)})+k} \left(\mathrm{err}_{i,k}^{\rho_i(n)} - \alpha \right).$$

Introduce the common reference sequence

$$\bar{\eta}_n := \eta_0 \left(1 + n\Delta_t + K\right)^{-a}, \qquad \lambda_n := \gamma_{t^{n+1}} \bar{\eta}_n.$$

The sequence $\bar{\eta}_n$ is used only to normalize the server increments; the actual local updates retain their observation-indexed step sizes. Dividing the preceding identity by λ_n yields

$$\frac{q^{n+1} - q^n}{\lambda_n} = E_n + R_n,$$

where

$$R_n := \sum_{i \in \mathcal{S}_n} d_i \sum_{k=1}^{K} \left(\frac{\eta_{n_i(t^{\rho_i(n)})+k}}{\bar{\eta}_n} - 1 \right) \left(\mathrm{err}_{i,k}^{\rho_i(n)} - \alpha \right).$$

Consequently,

$$\sum_{n=0}^{N-1} E_n = \sum_{n=0}^{N-1} \frac{q^{n+1} - q^n}{\lambda_n} - \sum_{n=0}^{N-1} R_n. \tag{18}$$

We first bound the term involving the global threshold increments. Under Assumption 3, λ_n is positive and nonincreasing. Define

$$\delta_0 := \lambda_0^{-1}, \qquad \delta_r := \lambda_r^{-1} - \lambda_{r-1}^{-1}, \qquad r \geq 1.$$

Then $\delta_r \geq 0$ and $\lambda_n^{-1} = \sum_{r=0}^{n} \delta_r$. Rearranging the finite sums gives

$$\begin{aligned}
\sum_{n=0}^{N-1} \frac{q^{n+1} - q^n}{\lambda_n} &= \sum_{n=0}^{N-1} \sum_{r=0}^{n} \delta_r \left(q^{n+1} - q^n\right) \\
&= \sum_{r=0}^{N-1} \delta_r \sum_{n=r}^{N-1} \left(q^{n+1} - q^n\right) \\
&= \sum_{r=0}^{N-1} \delta_r \left(q^N - q^r\right).
\end{aligned}$$

By Lemma 1, $|q^N - q^r| \leq C_q$. Therefore,

$$\left| \sum_{n=0}^{N-1} \frac{q^{n+1} - q^n}{\lambda_n} \right| \leq C_q \sum_{r=0}^{N-1} \delta_r = \frac{C_q}{\lambda_{N-1}}. \tag{19}$$

We next control the remainder arising from differences between local step sizes and the common reference sequence. Define

$$\mathcal{R}_N := \sum_{n=0}^{N-1} \sum_{i \in \mathcal{S}_n} d_i \sum_{k=1}^{K} \left| \frac{\eta_{n_i(t^{\rho_i(n)})+k}}{\bar{\eta}_n} - 1 \right|.$$

Since $|\mathrm{err}_{i,k}^{\rho_i(n)} - \alpha| \leq 1$, the triangle inequality implies

$$\left| \sum_{n=0}^{N-1} R_n \right| \leq \sum_{n=0}^{N-1} |R_n| \leq \mathcal{R}_N. \tag{20}$$

Combining Equation (18), Equation (19), and Equation (20) gives

$$\left| \sum_{n=0}^{N-1} E_n \right| \leq \frac{C_q}{\lambda_{N-1}} + \mathcal{R}_N. \tag{21}$$

Substitution into Equation (17) establishes the aggregation-weighted coverage bound:

$$|\mathrm{Cov}_d^{\mathrm{agg}}(N) - (1-\alpha)| \leq \frac{C_q/\lambda_{N-1} + \mathcal{R}_N}{\Lambda_N}.$$

We now extend this bound to all completed prediction–evaluation steps. Fix a system time $T \geq \Delta_t$, and let $N := \lfloor T/\Delta_t \rfloor$. Let $\mathcal{U}(T)$ denote the set of observation indices (i,t) whose prediction–evaluation steps have been completed by time T but whose task updates have not yet been aggregated. Define their weighted miscoverage deviation by

$$U(T) := \sum_{(i,t) \in \mathcal{U}(T)} d_i \left(\mathrm{err}_t^i - \alpha\right).$$

Partitioning the completed observations into aggregated and unaggregated tasks yields

$$\mathrm{Cov}_d(T) - (1-\alpha) = -\frac{1}{N_d(T)} \left[\sum_{n=0}^{N-1} E_n + U(T) \right], \qquad N_d(T) > 0. \tag{22}$$

Under the prescribed scheduling rule, each client has at most one task awaiting aggregation and hence at most K completed observations not yet represented in the server updates. Recalling $D := \sum_{i=1}^{M} d_i$, we obtain

$$|U(T)| \le \sum_{(i,t)\in\mathcal{U}(T)} d_i \le KD.$$

Moreover, the weighted observation counts satisfy

$$N_d(T) = \Lambda_N + \sum_{(i,t)\in\mathcal{U}(T)} d_i, \qquad \Lambda_N \le N_d(T) \le \Lambda_N + KD.$$

Applying Equation (21) and the bound on $U(T)$ to Equation (22) gives

$$|\mathrm{Cov}_d(T) - (1-\alpha)| \le \frac{C_q/\lambda_{N-1} + \mathcal{R}_N + KD}{N_d(T)}.$$

This proves the two finite-sample bounds with the remainder $\mathcal{R}_N$ defined above. We have completed the proof of this theorem. □

D.3 Proof of Theorem 2

Proof of Theorem 2. Let $\bar{\eta}_n := \eta_0(1 + n\Delta_t + K)^{-a}$. We know $\lambda_n := \gamma_{t^{n+1}}\bar{\eta}_n$. Notice that

$$0 \le \frac{\eta_{i,k}^{\rho_i(n)}}{\bar{\eta}_n} - 1 = \left(\frac{n+1}{1+\rho_i(n)}\right)^a - 1 = \left(1 + \frac{n-\rho_i(n)}{1+\rho_i(n)}\right)^a - 1$$

Under Assumptions 2 and 3, we then have

$$\frac{\eta_{i,k}^{\rho_i(n)}}{\bar{\eta}_n} - 1 \le \frac{a\big(n-\rho_i(n)\big)}{1+\rho_i(n)} \le \frac{a\tau}{1+(n-\tau)_+} \le \frac{a\tau(\tau+1)}{n+1}. \tag{23}$$

We therefore organize the proof into the following five steps: decompose the global recursion, control the cumulative perturbations, establish local–global synchronization, identify the drift over complete participation cycles, and prove convergence of the global and local thresholds.

Step 1. The algebraic decomposition in the proof of Theorem 1 gives

$$q^{n+1} - q^n = \lambda_n(E_n + R_n),$$
$$E_n := \sum_{i\in\mathcal{S}_n} d_i \sum_{k=1}^{K} \left(\mathrm{err}_{i,k}^{\rho_i(n)} - \alpha\right),$$

where R_n is the step-size remainder defined using the observation-indexed local updates.

By the filtration convention, the prediction rule is fixed before the corresponding response is revealed. Therefore,

$$\mathbb{E}\left[\mathrm{err}_{i,k}^{\rho_i(n)} \,\middle|\, \mathcal{F}_{i,k-1}^{\rho_i(n)}\right] = 1 - F_{i,k}^{\rho_i(n)}\left(q_{i,k-1}^{\rho_i(n)}\right).$$

Define the centered observation noise by

$$\xi_{i,k}^{\rho_i(n)} := \mathrm{err}_{i,k}^{\rho_i(n)} - \left[1 - F_{i,k}^{\rho_i(n)}\left(q_{i,k-1}^{\rho_i(n)}\right)\right].$$

Then

$$\mathbb{E}\left[\xi_{i,k}^{\rho_i(n)} \,\middle|\, \mathcal{F}_{i,k-1}^{\rho_i(n)}\right] = 0, \qquad \left|\xi_{i,k}^{\rho_i(n)}\right| \le 1.$$

Writing

$$g_n := \sum_{i\in\mathcal{S}_n} d_i \sum_{k=1}^{K} \left[(1-\alpha) - F_{i,k}^{\rho_i(n)}\left(q_{i,k-1}^{\rho_i(n)}\right)\right],$$
$$\xi_n := \sum_{i\in\mathcal{S}_n} d_i \sum_{k=1}^{K} \xi_{i,k}^{\rho_i(n)},$$

we obtain

$$q^{n+1} - q^n = \lambda_n g_n + \lambda_n \xi_n + \lambda_n R_n. \tag{24}$$

The martingale-difference property holds in observation-revelation order; it is not asserted for the delayed server-round aggregates ξ_n.

Step 2. The polynomial step-size schedules and $1/2 < a + b < 1$ imply

$$\lambda_n \longrightarrow 0, \qquad \sum_{n=0}^{\infty} \lambda_n = \infty, \qquad \sum_{n=0}^{\infty} \lambda_n^2 < \infty.$$

Moreover, λ_n is positive and nonincreasing. Let i_ℓ denote the client associated with the ℓth observation, n_ℓ the server round in which its task is aggregated, and ξ_ℓ^{obs} its centered noise. Since $d_{i_\ell}\lambda_{n_\ell}$ is $\mathcal{F}_{\ell-1}$-measurable,

$$\mathcal{M}_L := \sum_{\ell=1}^{L} d_{i_\ell}\lambda_{n_\ell}\xi_\ell^{\mathrm{obs}}$$

is a martingale with respect to $\{\mathcal{F}_L\}_{L\geq 0}$. Each task contains exactly K observations and is aggregated once, so

$$\begin{aligned}\sum_{\ell=1}^{\infty} d_{i_\ell}^2\lambda_{n_\ell}^2 &= K\sum_{n=0}^{\infty}\lambda_n^2\sum_{i\in\mathcal{S}_n} d_i^2 \\ &\leq K\left(\sum_{i=1}^{M} d_i^2\right)\sum_{n=0}^{\infty}\lambda_n^2 < \infty.\end{aligned}$$

Using $|\xi_\ell^{\mathrm{obs}}| \leq 1$ and the orthogonality of martingale differences gives

$$\sup_{L\geq 1}\mathbb{E}\big[\mathcal{M}_L^2\big] \leq K\left(\sum_{i=1}^{M} d_i^2\right)\sum_{n=0}^{\infty}\lambda_n^2 < \infty.$$

Thus, $\mathcal{M}_L$ converges almost surely.

The cumulative noise incorporated in the first N server aggregations is

$$Z_N := \sum_{n=0}^{N-1}\lambda_n\xi_n.$$

Let L_N be the number of observations revealed by the completion of the aggregation at t^N. The difference between $\mathcal{M}_{L_N}$ and Z_N consists only of observations whose task updates have not yet been aggregated. There are at most K such observations per client, and their aggregation-round indices are at least N. Hence, with $D := \sum_{i=1}^{M} d_i$,

$$|\mathcal{M}_{L_N} - Z_N| \leq KD\lambda_N \longrightarrow 0.$$

Since $L_N \to \infty$ under the fixed-duration schedule, Z_N converges almost surely.

The definition of R_n and Equation (23) further imply

$$|R_n| \leq \sum_{i\in\mathcal{S}_n} d_i \sum_{k=1}^{K}\left|\frac{\eta_{n_i(t^{\rho_i(n)})+k}}{\bar{\eta}_n} - 1\right| \leq \frac{KDC_\eta}{n+1}.$$

Consequently,

$$\sum_{n=0}^{\infty}\lambda_n|R_n| \leq KDC_\eta\sum_{n=0}^{\infty}\frac{\lambda_n}{n+1} < \infty.$$

It follows that

$$V_N := Z_N + \sum_{n=0}^{N-1}\lambda_n R_n$$

converges almost surely. In particular,

$$\lim_{N\to\infty} \sup_{N_2\geq N_1\geq N} |V_{N_2} - V_{N_1}| = 0 \qquad \text{almost surely.} \tag{25}$$

Step 3. By Equation (23), the expanded server recursion satisfies

$$\begin{aligned} |q^{n+1} - q^n| &\leq \lambda_n \sum_{i\in\mathcal{S}_n} d_i \sum_{k=1}^{K} \frac{\eta_{n_i(t^{\rho_i(n)})+k}}{\bar{\eta}_n} \\ &\leq KD(1+C_\eta)\lambda_n =: C\lambda_n, \end{aligned} \tag{26}$$

where C is independent of n.

For $i \in \mathcal{S}_n$, the bounded-staleness condition gives

$$\begin{aligned} \left|q_{i,k-1}^{\rho_i(n)} - q^n\right| &\leq \left|q^{\rho_i(n)} - q^n\right| + \left|q_{i,k-1}^{\rho_i(n)} - q^{\rho_i(n)}\right| \\ &\leq C \sum_{m=(n-\tau)_+}^{n-1} \lambda_m + \sum_{j=1}^{k-1} \eta_{n_i(t^{\rho_i(n)})+j} \\ &\leq C \sum_{m=(n-\tau)_+}^{n-1} \lambda_m + (K-1)\left(1 + \frac{C_\eta}{n+1}\right)\bar{\eta}_n \longrightarrow 0, \end{aligned}$$

where $(n-\tau)_+ := \max\{n-\tau, 0\}$. Because τ and K are fixed, the convergence is uniform over participating clients and local steps.

Step 4. Let $c_i := \lceil \tau_i/\Delta_t \rceil$, $W := \mathrm{lcm}(c_1, \ldots, c_M)$, where lcm denotes the least common multiple. Under the fixed-duration schedule, client i contributes once every c_i server rounds. Thus, each block of W rounds contains exactly W/c_i contributions from client i. Since $d_i = c_i p_i$,

$$\sum_{n=vW}^{(v+1)W-1} d_i \mathbf{1}\{i \in \mathcal{S}_n\} = Wp_i, \qquad v = 0, 1, \ldots. \tag{27}$$

Define the block step size and accumulated drift by

$$h_v := KW\lambda_{vW}, \qquad \Delta_v := \sum_{n=vW}^{(v+1)W-1} \lambda_n g_n.$$

Summing Equation (24) over one block yields

$$q^{(v+1)W} - q^{vW} = \Delta_v + V_{(v+1)W} - V_{vW}. \tag{28}$$

Moreover, monotonicity of λ_n and Equation (27) imply

$$|\Delta_v| \leq K\lambda_{vW} \sum_{n=vW}^{(v+1)W-1} \sum_{i\in\mathcal{S}_n} d_i = KW\lambda_{vW} = h_v.$$

The polynomial step-size schedules also give

$$\begin{aligned} h_v \longrightarrow 0, \qquad \sum_{v=0}^{\infty} h_v = \infty, \\ \max_{vW\leq n<(v+1)W} \left|\frac{\lambda_n}{\lambda_{vW}} - 1\right| \longrightarrow 0. \end{aligned}$$

Together with Equation (26) and the local–global bound established above, these properties imply

$$\max_{\substack{vW\leq n<(v+1)W \\ i\in\mathcal{S}_n,\ 1\leq k\leq K}} \left|q_{i,k-1}^{\rho_i(n)} - q^{vW}\right| \longrightarrow 0. \tag{29}$$

Fix a realization on which the score-distribution convergence, the global boundedness result, and Equation (25) hold. For any common continuity point u of $F_1, \dots, F_M$, Assumption 5 and Equation (27) give

$$\frac{1}{h_v} \sum_{n=vW}^{(v+1)W-1} \lambda_n \sum_{i \in \mathcal{S}_n} d_i \sum_{k=1}^{K} \left[(1-\alpha) - F_{i,k}^{\rho_i(n)}(u) \right] = (1-\alpha) - F(u) + o(1). \tag{30}$$

Indeed, the absolute difference between the normalized sum and $(1-\alpha) - F(u)$ is bounded by

$$\max_{vW \le n < (v+1)W} \left| \frac{\lambda_n}{\lambda_{vW}} - 1 \right| + \max_{\substack{vW \le n < (v+1)W \\ i \in \mathcal{S}_n,\ 1 \le k \le K}} \left| F_{i,k}^{\rho_i(n)}(u) - F_i(u) \right|,$$

and both terms converge to zero.

Fix $\varepsilon > 0$. A finite collection of distribution functions has at most countably many discontinuity points. We may therefore choose common continuity points

$$u_- \in \left(q^* - \frac{\varepsilon}{2}, q^* \right), \qquad u_+ \in \left(q^*, q^* + \frac{\varepsilon}{2} \right).$$

Assumption 6 ensures that

$$\delta_\varepsilon := \frac{1}{2} \min \left\{ (1-\alpha) - F(u_-), F(u_+) - (1-\alpha) \right\} > 0.$$

If $q^{vW} \ge q^* + \varepsilon$, then Equation (29) implies that every local threshold used in the block exceeds u_+ for all sufficiently large v. Monotonicity of the conditional distribution functions and Equation (30) consequently yield

$$\frac{\Delta_v}{h_v} \le (1-\alpha) - F(u_+) + o(1) \le -\delta_\varepsilon.$$

Similarly, if $q^{vW} \le q^* - \varepsilon$, then

$$\frac{\Delta_v}{h_v} \ge (1-\alpha) - F(u_-) + o(1) \ge \delta_\varepsilon.$$

Thus, for all sufficiently large v,

$$\begin{aligned} q^{vW} \ge q^* + \varepsilon \quad &\Longrightarrow \quad \Delta_v \le -\delta_\varepsilon h_v, \\ q^{vW} \le q^* - \varepsilon \quad &\Longrightarrow \quad \Delta_v \ge \delta_\varepsilon h_v. \end{aligned} \tag{31}$$

Step 5. Fix $\varepsilon > 0$. Choose v_0 sufficiently large that Equation (31) holds with $\varepsilon/2$ in place of ε and

$$h_v \le \frac{\varepsilon}{8}, \qquad v \ge v_0,$$
$$\sup_{s \ge r \ge v_0} |V_{sW} - V_{rW}| \le \frac{\varepsilon}{8}.$$

By Equation (28),

$$|q^{(v+1)W} - q^{vW}| \le \frac{\varepsilon}{4}, \qquad v \ge v_0.$$

We first show that the block sequence enters $[q^* - \varepsilon/2, q^* + \varepsilon/2]$. If it never enters this interval after v_0, the preceding increment bound prevents it from moving from one side to the other without entering. It must therefore remain on one side. If it remains above $q^* + \varepsilon/2$, then Equation (28) and Equation (31) imply

$$q^{vW} - q^{v_0 W} \le -\delta_{\varepsilon/2} \sum_{j=v_0}^{v-1} h_j + \frac{\varepsilon}{8} \longrightarrow -\infty,$$

contradicting Lemma 1. Remaining below $q^* - \varepsilon/2$ similarly gives

$$q^{vW} - q^{v_0 W} \geq \delta_{\varepsilon/2} \sum_{j=v_0}^{v-1} h_j - \frac{\varepsilon}{8} \longrightarrow +\infty,$$

which is also impossible. Hence, there exists $v_1 \geq v_0$ such that $|q^{v_1 W} - q^*| \leq \varepsilon/2$.

We next show that the block sequence cannot subsequently leave $[q^* - \varepsilon, q^* + \varepsilon]$. Suppose that $q^{wW} > q^* + \varepsilon$ for some $w > v_1$, and define

$$r := \max\left\{ m \in \{v_1, \ldots, w-1\} : q^{mW} \leq q^* + \frac{\varepsilon}{2} \right\}.$$

This set is nonempty. By maximality of r, $q^{jW} > q^* + \varepsilon/2$ for $r < j < w$, so Equation (31) gives $\Delta_j \leq 0$ on these blocks. Using $|\Delta_r| \leq h_r$ and Equation (28), we obtain

$$\begin{aligned} q^{wW} - q^{rW} &= \sum_{j=r}^{w-1} \Delta_j + V_{wW} - V_{rW} \\ &\leq h_r + \frac{\varepsilon}{8} \leq \frac{\varepsilon}{4}. \end{aligned}$$

Therefore, $q^{wW} \leq q^* + 3\varepsilon/4$, contradicting the assumed upper escape.

For a lower escape, suppose that $q^{wW} < q^* - \varepsilon$ and instead define

$$r := \max\left\{ m \in \{v_1, \ldots, w-1\} : q^{mW} \geq q^* - \frac{\varepsilon}{2} \right\}.$$

Then $\Delta_j \geq 0$ for $r < j < w$, and

$$q^{wW} - q^{rW} \geq -h_r - \frac{\varepsilon}{8} \geq -\frac{\varepsilon}{4}.$$

It follows that $q^{wW} \geq q^* - 3\varepsilon/4$, again a contradiction. Thus, $|q^{vW} - q^*| \leq \varepsilon$ for every $v \geq v_1$. Since $\varepsilon > 0$ is arbitrary, $q^{vW} \to q^*$.

To extend convergence to all server rounds, let $vW \leq n < (v+1)W$. By Equation (26),

$$\begin{aligned} |q^n - q^*| &\leq |q^{vW} - q^*| + \sum_{m=vW}^{n-1} |q^{m+1} - q^m| \\ &\leq |q^{vW} - q^*| + CW\lambda_{vW} \longrightarrow 0. \end{aligned}$$

Hence, $q^n \to q^*$ almost surely.

Finally, let $q^i(T)$ denote the threshold stored by client i at system time T, and let $r_i(T)$ be the initialization-round index of its current task. Under the fixed-duration protocol, both $r_i(T)$ and $n_i(t^{r_i(T)})$ tend to infinity as $T \to \infty$. If the client has completed $J_i(T) \leq K$ local updates in this task, then

$$\begin{aligned} |q^i(T) - q^*| &\leq |q^{r_i(T)} - q^*| + |q^i(T) - q^{r_i(T)}| \\ &\leq |q^{r_i(T)} - q^*| + \sum_{k=1}^{J_i(T)} \eta_{n_i(t^{r_i(T)})+k} \\ &\leq |q^{r_i(T)} - q^*| + K\eta_{n_i(t^{r_i(T)})+1} \longrightarrow 0. \end{aligned}$$

Since the number of clients is finite,

$$\max_{1 \leq i \leq M} |q^i(T) - q^*| \longrightarrow 0 \qquad \text{almost surely as } T \to \infty.$$

This completes the proof under Equation (23). The proof of this theorem have been completed. □

D.4 Proof of Corollary 1

Proof of Corollary 1. We compare the conditional score distributions at fixed continuity points on either side of q^*, following the argument of Angelopoulos et al. (2024). By Theorem 2 and the uniform local–global discrepancy established in Step 3 of its proof,

$$\begin{aligned}&\max_{\substack{i\in\mathcal{S}_n\\1\le k\le K}}\left|q_{i,k-1}^{\rho_i(n)}-q^*\right|\\&\quad\le\max_{\substack{i\in\mathcal{S}_n\\1\le k\le K}}\left|q_{i,k-1}^{\rho_i(n)}-q^n\right|+|q^n-q^*|\longrightarrow 0\qquad\text{almost surely.}\end{aligned}\tag{32}$$

Throughout the proof, maxima over an empty participating set are defined as zero. Fix a sample path on which Equation (32) and Assumption 5 hold.

Let $\varepsilon>0$. Since the number of clients is finite and each F_i is continuous at q^*, we can choose common continuity points $u_-<q^*<u_+$ of $F_1,\dots,F_M$ such that

$$\max_{1\le i\le M}\{F_i(q^*)-F_i(u_-)\}<\frac{\varepsilon}{2},$$
$$\max_{1\le i\le M}\{F_i(u_+)-F_i(q^*)\}<\frac{\varepsilon}{2}.$$

Such points exist arbitrarily close to q^* because the union of the discontinuity sets of $F_1,\dots,F_M$ is at most countable.

By Equation (32), for all sufficiently large n,

$$u_-<q_{i,k-1}^{\rho_i(n)}<u_+,\qquad i\in\mathcal{S}_n,\quad k=1,\dots,K.$$

At these fixed comparison points, Assumption 5 and the finiteness of the client set also imply that, for all sufficiently large n,

$$\max_{\substack{i\in\mathcal{S}_n\\1\le k\le K}}\left|F_{i,k}^{\rho_i(n)}(u)-F_i(u)\right|<\frac{\varepsilon}{2},\qquad u\in\{u_-,u_+\}.$$

Using Equation (4) and the monotonicity of the conditional distribution functions, we obtain

$$\begin{aligned}\mathrm{Coverage}_{i,k}^{\rho_i(n)}&\ge F_{i,k}^{\rho_i(n)}(u_-)>F_i(u_-)-\frac{\varepsilon}{2}>F_i(q^*)-\varepsilon,\\\mathrm{Coverage}_{i,k}^{\rho_i(n)}&\le F_{i,k}^{\rho_i(n)}(u_+)<F_i(u_+)+\frac{\varepsilon}{2}<F_i(q^*)+\varepsilon.\end{aligned}$$

These bounds hold simultaneously for every participating client and every local step. Since $\varepsilon>0$ is arbitrary, they establish Equation (7). In particular, the argument uses distributional convergence only at the fixed points u_- and u_+, avoiding substitution of a random threshold into a pointwise limit.

It remains to establish the client-weighted coverage limit. Continuity of each F_i at q^* implies continuity of $F=\sum_{i=1}^M p_iF_i$ at the same point. Taking $\delta\downarrow 0$ in the strict-crossing condition of Assumption 6 yields

$$\sum_{i=1}^M p_iF_i(q^*)=F(q^*)=1-\alpha.$$

Under the fixed-duration scheduling protocol, client i contributes once every $c_i:=\lceil\tau_i/\Delta_t\rceil$ server rounds. Its most recent participating round among the first N rounds therefore satisfies

$$\nu_i(N)=c_i\left\lfloor\frac{N}{c_i}\right\rfloor-1\longrightarrow\infty,\qquad N\ge c_i.$$

Thus, applying Equation (7) along $n=\nu_i(N)$ gives

$$\max_{1\le k\le K}\left|\mathrm{Coverage}_{i,k}^{\rho_i(\nu_i(N))}-F_i(q^*)\right|\longrightarrow 0$$

for every client i. By the definition of $\mathrm{Coverage}_{p,k}(N)$ and the triangle inequality,

$$\begin{aligned}
&\max_{1\le k\le K} \left|\mathrm{Coverage}_{p,k}(N) - (1-\alpha)\right| \\
&\quad = \max_{1\le k\le K} \left|\sum_{i=1}^{M} p_i \left[\mathrm{Coverage}_{i,k}^{\rho_i(\nu_i(N))} - F_i(q^*)\right]\right| \\
&\quad \le \sum_{i=1}^{M} p_i \max_{1\le k\le K} \left|\mathrm{Coverage}_{i,k}^{\rho_i(\nu_i(N))} - F_i(q^*)\right| \longrightarrow 0.
\end{aligned}$$

The last convergence follows because M is finite. Since the argument holds on an event of probability one, this proves Equation (8) and completes the proof. □

E Additional Experimental Results

This appendix presents additional figures and tables for the simulation and real-data experiments in Section 5. Tables 1 and 2 summarize the average performance on the synthetic and real-world datasets, respectively, across different distributional settings and levels of hardware heterogeneity. Figure 6 reports the regression results under F30 and F70 hardware heterogeneity for Example 1. Figures 7–9 present the corresponding classification results for Example 2. For the real-world experiments in Example 3, Figure 10 reports the results on DJIA Stocks and Beijing Air under F30 and F70, while Figures 11–13 report the results on MNIST, Fashion-MNIST, CIFAR-10, and WESAD under F30, F70 and F90.

As shown in Table 1, OAFCP maintains coverage close to the target level across all three regression settings. Its advantage is most pronounced under abrupt distributional changes, where it achieves lower loss and narrower prediction intervals than OSFCP. Under gradual drift, OAFCP similarly yields lower loss and tighter intervals, whereas the two methods perform comparably in the stationary setting. Figure 6 further shows that these improvements become more pronounced as client-side hardware heterogeneity increases. For classification, Figures 7–9 show that OAFCP and OSFCP achieve similar coverage across the considered settings, with relatively small differences in loss and prediction-set size. Overall, OAFCP remains competitive for classification while exhibiting greater robustness and interval efficiency for regression under distribution shift.

The real-data results show a similar pattern. As reported in Table 2 and Figure 10, OAFCP maintains coverage closer to the target level on both regression datasets. In contrast, OSFCP tends to over-cover on DJIA Stocks and under-cover on Beijing Air. OAFCP also achieves lower loss and substantially narrower intervals on DJIA Stocks, while producing comparable interval widths on Beijing Air. On the four classification datasets, Figures 11–13 show broadly comparable coverage between the two methods. OSFCP performs slightly better on WESAD, which exhibits slightly conservative intervals and slightly higher coverage rates under the F70 and F90 scenarios. Whereas OAFCP performs slightly better on Fashion-MNIST and CIFAR-10 in terms of coverage. OAFCP may incur higher loss and larger prediction sets on more challenging datasets such as CIFAR-10, but achieves notably lower loss on WESAD. Overall, the results indicate that OAFCP provides more reliable calibration on the real-world regression tasks while remaining competitive across the classification benchmarks.

Table 1: Performance comparison across regression and classification settings on synthetic setting. $T = 1000, \Delta_t = 2, K = 10, M = 10$. We report three metrics: average coverage, average loss and average interval size.

Method	Average coverage			Average loss			Average size		
	F30	F70	F90	F30	F70	F90	F30	F70	F90
regression									
setting 1: Stationary Distribution									
OAFCP	0.9040	0.9032	0.9028	0.0122	0.0111	0.0118	0.4654	0.4263	0.4096
OSFCP	0.9076	0.9081	0.9088	0.0123	0.0143	0.0180	0.4365	0.4448	0.4573
setting 2: Abrupt Changes									
OAFCP	0.9054	0.9047	0.9044	0.0800	0.0765	0.0753	0.7883	0.7395	0.7173
OSFCP	0.8917	0.8923	0.8927	0.1946	0.1964	0.2000	1.2025	1.2100	1.2236
setting 3: Gradual Drift									
OAFCP	0.9037	0.9030	0.9029	0.0131	0.0120	0.0127	0.4931	0.4542	0.4372
OSFCP	0.9035	0.9040	0.9047	0.0196	0.0217	0.0253	0.6047	0.6130	0.6251
classification									
setting 1: Stationary Distribution									
OAFCP	0.9005	0.9007	0.9002	0.9563	0.9544	0.9532	2.3050	2.2895	2.2755
OSFCP	0.9004	0.9004	0.9004	0.9521	0.9519	0.9529	2.2701	2.2684	2.2703
setting 2: Gradual Drift									
OAFCP	0.9005	0.9006	0.9003	0.9514	0.9488	0.9482	2.3220	2.3026	2.2886
OSFCP	0.9006	0.9005	0.9005	0.9521	0.9516	0.9524	2.2876	2.2868	2.2865
setting 3: Class Emergence									
OAFCP	0.9006	0.9007	0.9004	0.9924	0.9898	0.9879	2.5059	2.4664	2.4349
OSFCP	0.9005	0.9004	0.9005	0.9897	0.9907	0.9914	2.4297	2.4311	2.4310

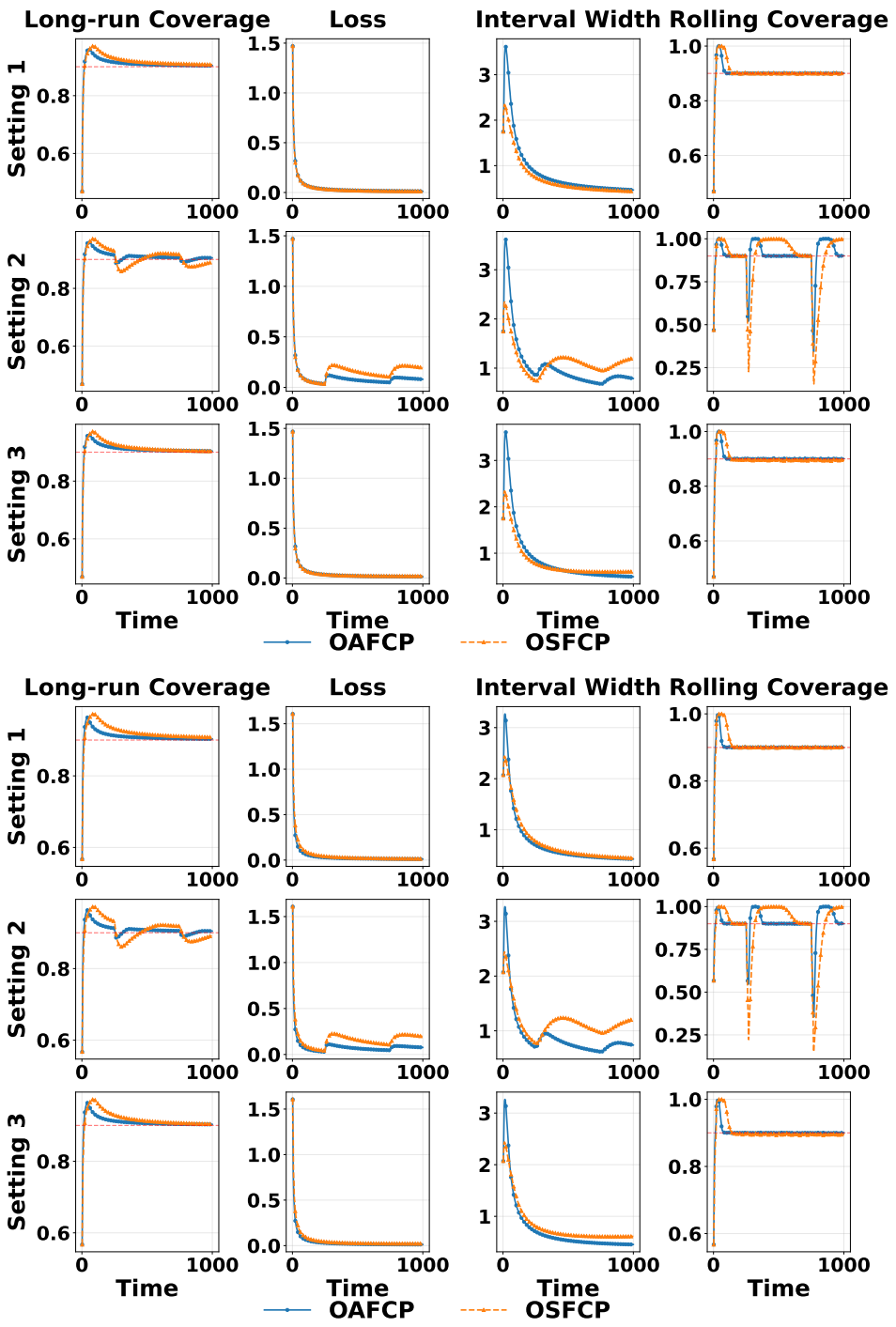


Figure 6: Simulation regression results under F30 (up) and F70 (down).

Table 2: Performance comparison across regression and classification datasets on 6 real datasets.$T = 1000, \Delta_t = 2, K = 10$. We report three metrics: average coverage, average loss and average interval size.

Dataset	Method	Average coverage			Average loss			Average size		
		F30	F70	F90	F30	F70	F90	F30	F70	F90
regression										
DJIA Stocks										
	OAFCP	0.9111	0.9032	0.8992	0.0035	0.0079	0.0203	0.1726	0.1623	0.1502
	OSFCP	0.9724	0.9726	0.9738	0.0080	0.0139	0.0380	0.5773	0.5738	0.5800
Beijing Air										
	OAFCP	0.9089	0.9066	0.9088	0.0885	0.0781	0.0721	1.3228	1.2313	1.1877
	OSFCP	0.8761	0.8764	0.8767	0.0894	0.0895	0.0895	1.2412	1.2418	1.2424
classification										
MNIST										
	OAFCP	0.8987	0.8997	0.8940	0.1763	0.1874	0.2162	0.9239	0.9198	0.9180
	OSFCP	0.8983	0.8982	0.8981	0.1718	0.1720	0.1720	0.9153	0.9153	0.9152
Fashion-MNIST										
	OAFCP	0.9010	0.9014	0.9026	0.2731	0.2496	0.2285	0.9521	0.9311	0.9311
	OSFCP	0.8969	0.8969	0.8965	0.2398	0.2406	0.2405	0.9263	0.9275	0.9275
CIFAR-10										
	OAFCP	0.9006	0.8974	0.8969	0.6376	0.4919	0.3633	0.9879	0.9430	0.9246
	OSFCP	0.8971	0.8968	0.8963	0.3079	0.3117	0.3123	0.9122	0.9122	0.9112
WESAD										
	OAFCP	0.8991	0.8894	0.8920	0.0160	0.0155	0.0142	0.9205	0.8915	0.8943
	OSFCP	0.9052	0.9052	0.9052	0.0316	0.0313	0.0311	0.9064	0.9065	0.9067

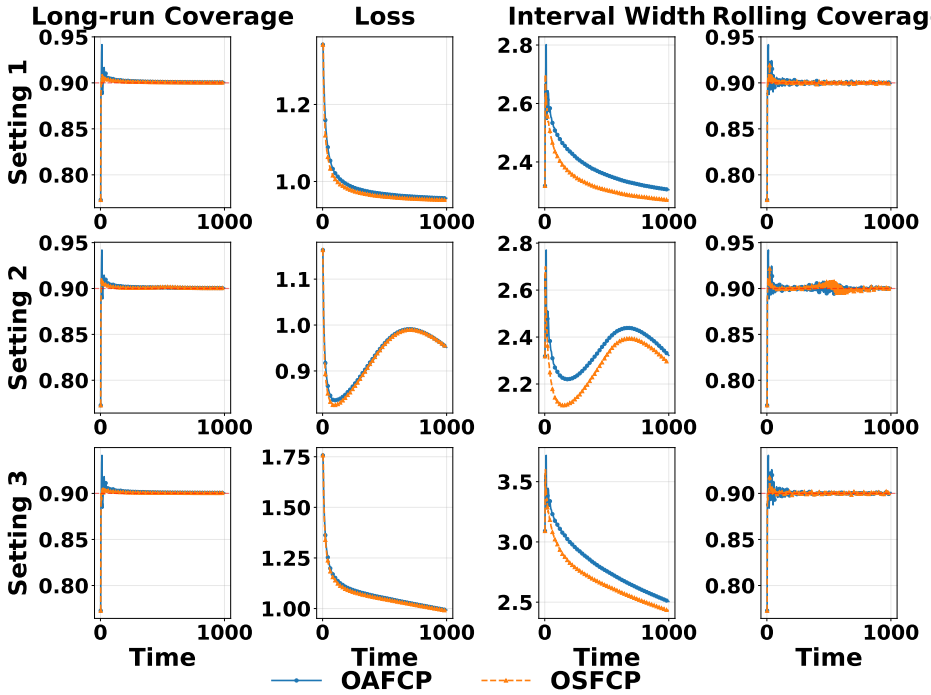


Figure 7: Simulation classification results under F30.

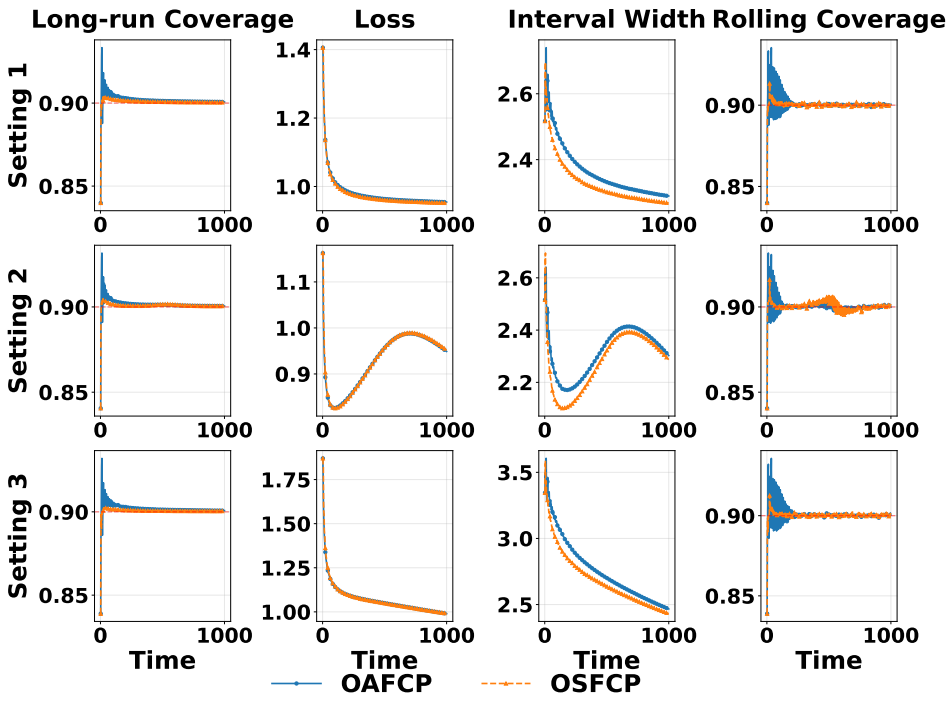


Figure 8: Simulation classification results under F70.

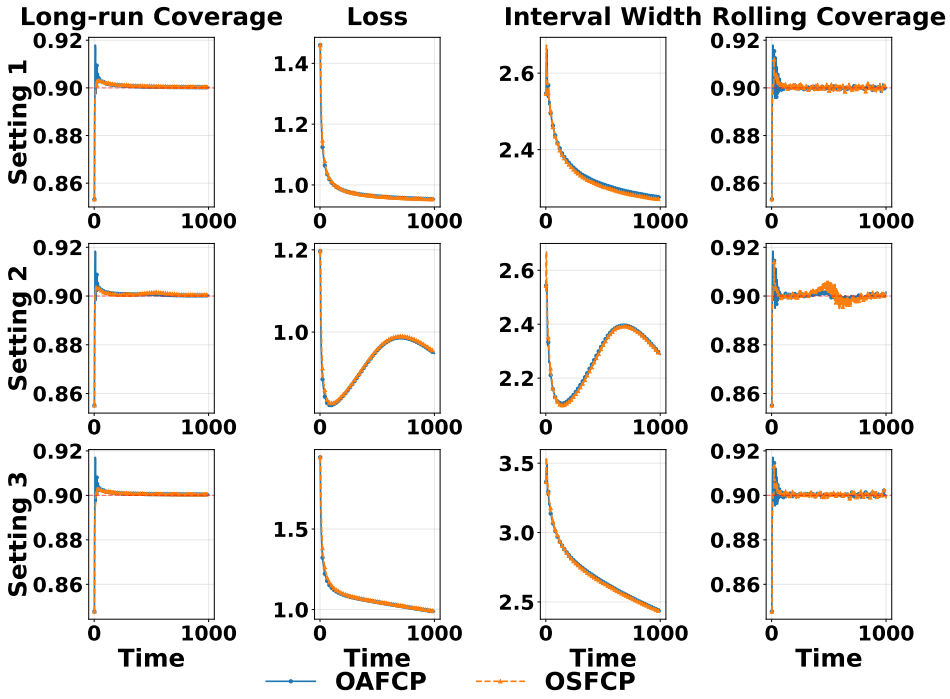


Figure 9: Simulation classification results under F90.

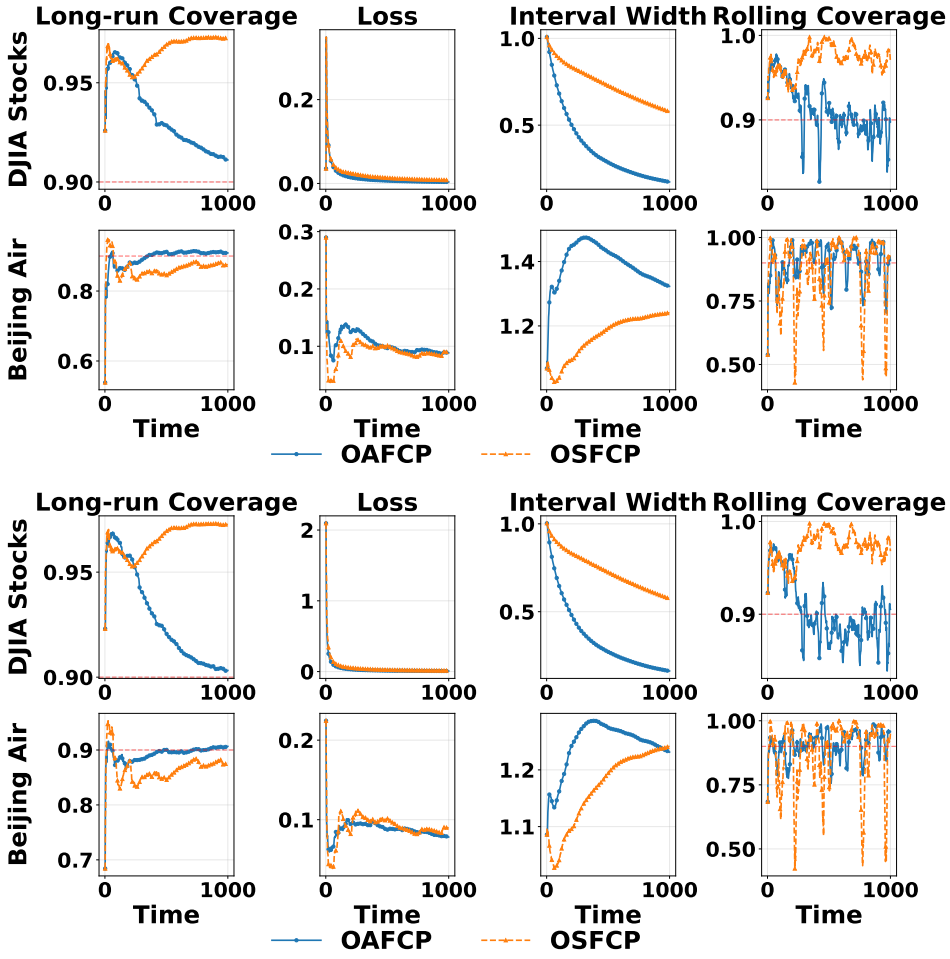


Figure 10: Simulation results based on two real-world regression datasets under F30 (up) and F70 (down) conditions.

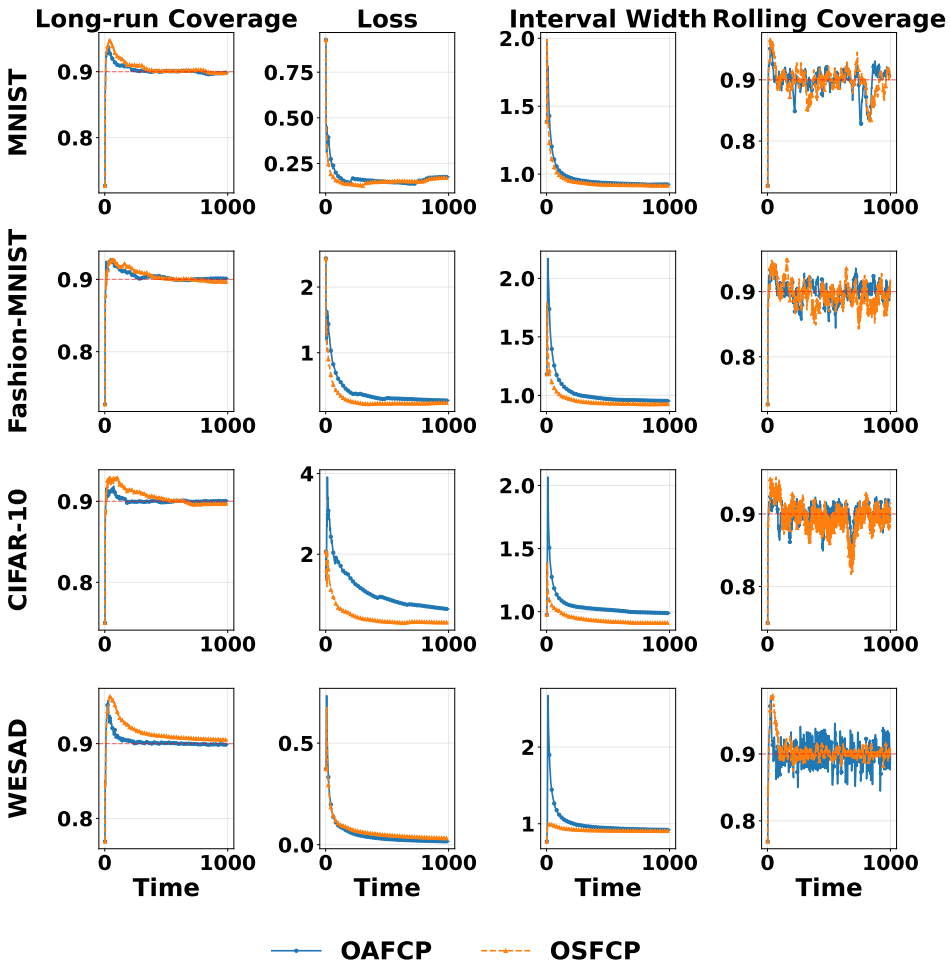


Figure 11: Simulation results based on four real-world classification datasets under F30 condition.

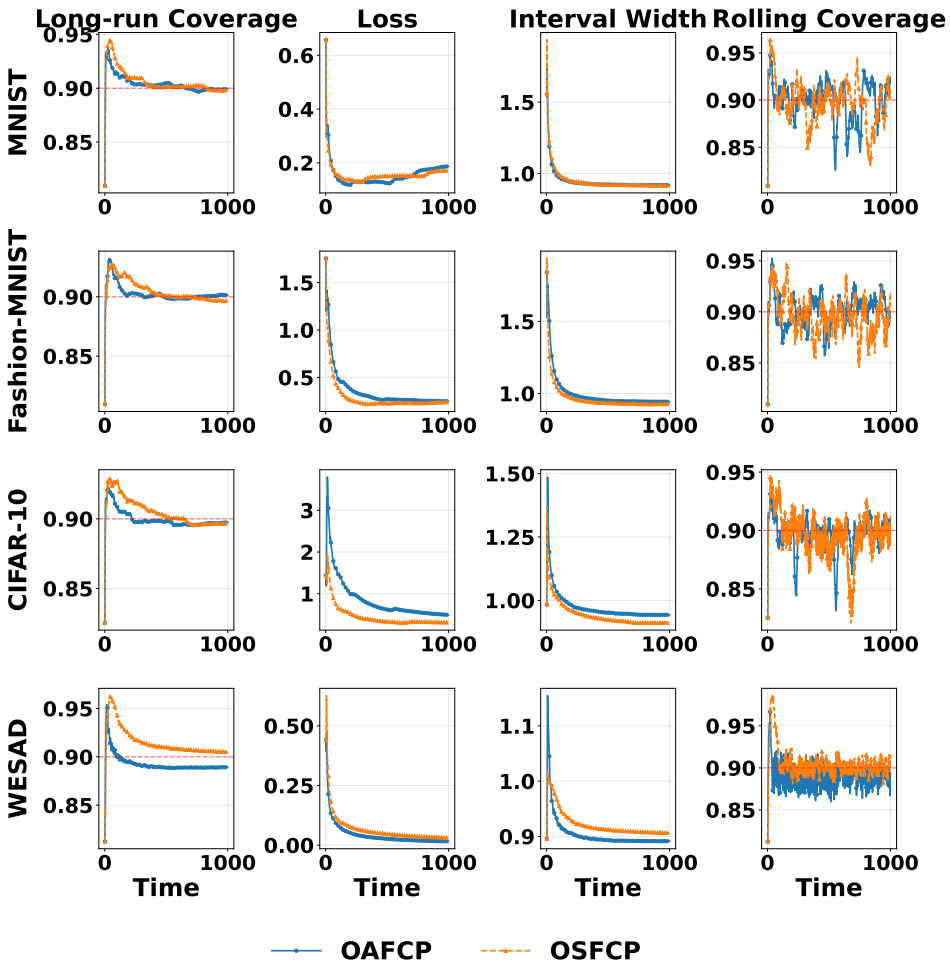


Figure 12: Simulation results based on four real-world classification datasets under F70 condition.

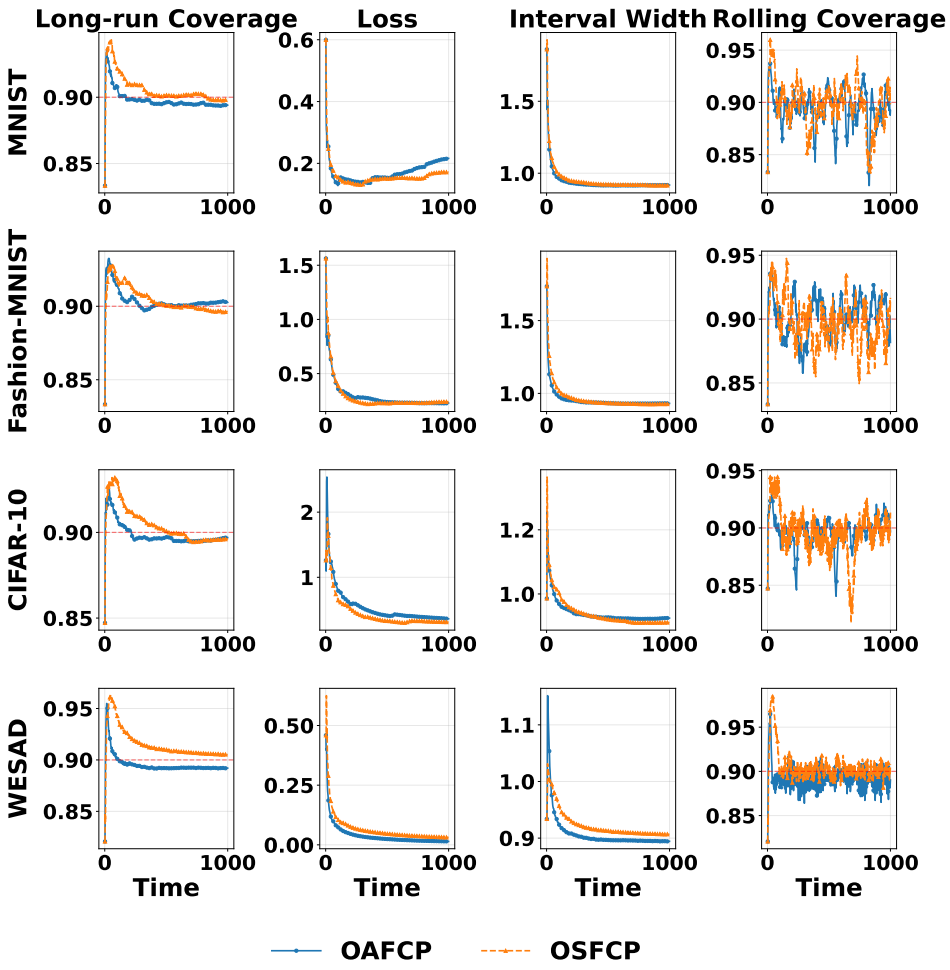


Figure 13: Simulation results based on four real-world classification datasets under F90 condition.