POLITECNICO DI TORINO
Repository ISTITUZIONALE

Characterizing the performance of classification models through conformal correlation matrices

Original
Characterizing the performance of classification models through conformal correlation matrices / Perlo, A., Chiasserini, C.F., De Veciana, G., Malandrino, F.. - In: COMPUTER COMMUNICATIONS. - ISSN 0140-3664. - 247:(2026). [10.1016/j.comcom.2025.108398]

Availability:
This version is available at: 11583/3005667 since: 2025-12-05T14:18:56Z

Publisher:
Elsevier

Published
DOI:10.1016/j.comcom.2025.108398

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

26 November 2026

Computer Communications 247 (2026) 108398



Contents lists available at ScienceDirect

Computer Communications

journal homepage: www.elsevier.com/locate/comcom



Characterizing the performance of classification models through conformal correlation matrices☆

Alessandro Perlo [a], Carla Fabiana Chiasserini [a,b,c,*], Gustavo De Veciana [d], Francesco Malandrino [c,b]

[a] Politecnico di Torino, Torino, Italy
[b] CNIT, Italy
[c] CNR-IEIIT, Torino, Italy
[d] University of Austin, TX, USA

ARTICLE INFO

Keywords:
Distributed learning
Conformal prediction

ABSTRACT

In classification tasks, it is critical to accurately distinguish between specific classes, as misclassifications can undermine system reliability and user trust. In this paper, we study how client selection in both centralized and federated learning environments affects the performance of classification models trained on heterogeneous data. When training datasets across clients are statistically diverse, careful client selection becomes crucial to improve the ability of the model to discriminate between classes, while preserving privacy. In particular, we introduce a novel metric based on conformal prediction outcomes – the conformal correlation matrix – which captures the likelihood of class pairs co-occurring within conformal prediction sets. Unlike the traditional confusion matrix, which quantifies actual misclassifications, our metric characterizes potential ambiguities between classes, thus offering a complementary perspective on model performance and uncertainty. Through a series of examples, we demonstrate how our proposed metric can guide informed client selection and enhance model performance in both centralized and federated training settings. Our results highlight the potential of conformal-based metrics to improve classification reliability while safeguarding sensitive information about individual client data.

1. Introduction

Customizing Machine Learning (ML) models for specific user needs is essential to ensure that their predictions align with the particular context in which they are applied. Generic models, while powerful, often fail to capture the nuances of individual datasets or decision-making priorities. In the case of classification tasks, it is in particular important the model's ability to accurately distinguish between specific classes. Misclassification between closely related categories—such as between normal and critical network states, can undermine the system's reliability and user confidence.

In this paper we consider both centralized and decentralized, e.g., federated learning, settings where multiple clients and their associated data sets are used to train an ML model for a given classification task. Due to cost, privacy, or communication overheads constraints, it is of interest to limit the number of clients participating in the training process. If the clients' training data sets are statistically homogeneous, e.g., I.I.D., and of equal size, decisions on limiting the number of clients participating in the training process are straightforward, as there is a natural "exchangeability" among clients that would suggest one can simply choose subsets of clients at random or use a round robin type approach to encourage all of clients' data to contribute equally to the training process. When the clients' data sets are heterogeneous the selection of the clients participating across rounds of the training process might be done more judiciously and this has indeed been a topic of intense interest and research.

In our study, we propose a new approach at getting a better understanding of how client selection may affect the training process with a view on achieving improved performance. The performance of a trained classifier is usually evaluated based on its associated confusion matrix, which includes information on its accuracy for various classes as well as the likelihood of confusing one class with another. Based on the confusion matrix one can recover various relevant metrics, such as accuracy, recall and precision. Further, one can use the confusion matrix to see possibly diagnose issues like class imbalance, i.e., imbalance in the

☆ This article is part of a Special issue entitled: 'Arturo Azcorra' published in Computer Communications.

* Corresponding author at: Politecnico di Torino, Torino, Italy.
E-mail address: carla.chiasserini@polito.it (C.F. Chiasserini).

https://doi.org/10.1016/j.comcom.2025.108398
Received 21 October 2025; Received in revised form 2 December 2025; Accepted 4 December 2025
Available online 4 December 2025
0140-3664/© 2025 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

class representation in the training dataset which may bias the model towards classes that are more prominent. This type of information can then be used to potentially guide further client selection and model training.

Our contributions. In this paper we introduce a new metric which is based on the outcomes of conformal prediction for a given classification model and input data. Conformal prediction sets are random sets that, for a given input, are guaranteed to contain the true class with a pre-specified probability, i.e., a given confidence level. We capture the structure of these random sets, based on the likelihood that pairs of classes jointly occur in the conformal prediction sets. Specifically, we evaluate a correlation matrix associated with co-occurrence of pairs of classes in conformal sets — referred to as the conformal correlation matrix.

We show that CCMs can be integrated into the training process in two main ways. In centralized settings, they allow one to gauge how performance could be improved, e.g., which types of *potential errors* or ambiguity across classes have the potential to impair the future evolution of training. In distributed settings, CCMs can be integrated within the learning orchestration loop, driving node selection while protecting information that might best be kept private regarding individual client's performance.

Paper organization. In Section 3, we briefly introduce background on federated learning and conformal prediction, followed by Section 4 where we introduce the conformal correlation matrix. Section 5 discusses centralized experimental settings which were used to explore three potential benefits of conformal correlation matrices towards characterizing the performance of an ML model. Further, Section 6 extends the discussion of such benefits to a federated learning setting where one might be particularly concerned with privacy and the selection of clients so as to improve training performance. Finally, Section 7 summarizes our findings and outlines future research directions.

2. Related work

Conformal prediction is an uncertainty quantification framework described in Section 3.2. Beyond its role in measuring uncertainty, [1] leverage conformal prediction for uncertainty-aware optimization by incorporating prediction set size directly into the loss function. In addition, they integrate the *coverage confusion matrix* into the objective to penalize systematic overlaps where data from certain classes are grouped together.

Federated learning is a decentralized machine learning paradigm described in Section 3.1. Conformal prediction has been extended to the federated learning setting using different approaches. For example, [2] employ a t-Digest data structure to enable efficient communication and aggregation of approximate quantiles across clients, and provide a characterization of coverage guarantees under this approximation. Other approaches include the use of the pinball loss to estimate quantiles in a federated manner [3], and the computation of the conformity score quantile via a quantile-of-quantiles strategy [4], both of which are accompanied by appropriate theoretical coverage guarantees.

In federated learning we distinguish between *system heterogeneity*, that is, differences in device capabilities such as computation, communication bandwidth, and availability, and *data heterogeneity* [5], that is, non-identical and non-independent (non-IID) data distributions across clients. These heterogeneities motivate client selection strategies, which typically target *statistical utility*, arising from data heterogeneity, and *system utility*, arising from system heterogeneity.

Relevant contributions that incorporate statistical utility, sometimes in combination with system utility, include the following. Early contributions, such as [6,7], rely on training-time loss as a proxy for statistical utility to guide the selection of participating clients. In contrast, other approaches focus on gradient-based metrics. For instance, [8,9] utilize properties of client gradients, such as their norm or alignment with the global update direction, to identify clients that are likely to contribute beneficial updates to the global model. Client selection has also been explored as a mechanism to mitigate the impact of malicious or low-quality clients. Notably, [10] leverage conformal prediction to filter out potentially corrupt clients.

3. Background

In this section, we review concepts underlying federated learning and conformal prediction which we will use in defining Conformal Correlation Matrices (CCMs) and articulating how to use them.

3.1. Federated learning

The goal of federated learning (FL) [11] is to harness the possibly private data set and computational power of multiple learning nodes, hereinafter referred to as *clients*, to train one (or more) ML models mediated by an *FL server*. For concreteness, in the following we consider the ML model being trained to be a classifier. FL operates as follows:

1. each client performs one or more local training epochs on its model using its local dataset;
2. clients upload their model to the FL server;
3. the FL server combines the clients' model parameters, obtaining a global model;
4. the FL server sends the global model to the clients;
5. local training at the clients resumes using the global model.

We consider a set of clients $\mathcal{N}$ (with cardinality $N = |\mathcal{N}|$) who are available to participate in the training process. Each client $i \in \mathcal{N}$ possesses a local dataset $\mathcal{D}_i = \{(x_{i,j}, y_{i,j})\}_{j=1}^{n_i}$, whose size is $d_i = |D_i|$. Elements $x_{i,j}$ and $y_{i,j}$ in the datasets correspond, respectively, to samples (e.g., images) and labels (e.g., classes). We let $\mathbf{w}_i^k$ denote the local model parameters of client i at round k.

The goal of FL is [11] to minimize loss function f, over a global model $\mathbf{w}$, with f being defined as an average of local loss functions $f_i(\mathbf{w})$, each computed by running the global model $\mathbf{w}$ over local dataset D_i:

$$f(\mathbf{w}) = \frac{1}{N} \sum_{i \in \mathcal{N}} f_i(\mathbf{w}).$$

The learning process takes place over multiple *rounds* $k \in \{1, \dots, K\}$. At each round k, $\mathbf{w}_i^k$ represents the local model at client $i \in \mathcal{N}$ after that client performs its local training epoch(s). Then, global models $\mathbf{w}^{k+1}$ are obtained by combining local models according to a specific *aggregation strategy*. Aggregation strategies consist of two main decisions:

- which clients $\mathcal{P}^k \subseteq \mathcal{N}$ to use at each round k;
- how to combine the local models $\mathbf{w}_i^k$ into the global one $\mathbf{w}^{k+1}$.

The second issue can be generalized as setting the weights p_i^k to use in the relation:

$$\mathbf{w}^{k+1} \leftarrow \frac{1}{\sum_{j \in \mathcal{N}} p_j^k} \sum_{i \in \mathcal{N}} p_i^k \mathbf{w}_i^k,$$

where $p_i^k = 0$ for clients that are not selected at round k, i.e., $i \notin \mathcal{P}^k \Longrightarrow p_i^k = 0$. Importantly, selected nodes can be associated with any value of p_i, and p_i values may be different for different nodes.

The simplest approach is FedAvg, also used in the original work [11], which averages all local parameters, hence, is equivalent to setting $p_i^k = \frac{1}{|\mathcal{P}^k|}$:

$$\mathbf{w}^{k+1} \leftarrow \frac{1}{|\mathcal{P}^k|} \sum_{i \in \mathcal{P}^k} \mathbf{w}_i^k.$$

Later FL variants consider more complex ways of combining local updates. As an example, the FL server can decide to weight local

updates proportionally to the sizes $|D_i|$ of the corresponding local datasets, i.e., set $p_i^k = d_i$, obtaining:

$$\mathbf{w}^{k+1} \leftarrow \frac{1}{\sum_{i\in\mathcal{P}^k} d_i} \sum_{i\in\mathcal{P}^k} d_i \mathbf{w}_i^k.$$

Regardless of how updates are combined, *client selection* is arguably the most consequential decision in FL, with significant impact on the training time duration, resource consumption at the clients, and the quality level of the trained ML model. This is especially critical when different clients have local datasets of different *quality*, e.g., with different label distributions — also known as *non-i.i.d.* datasets. Examples of client selection schemes include power-of-choice [7], where clients with larger local losses are more likely to be selected, and Oort [6], combining local losses, level of staleness (i.e., how recent local datasets are), and client availability.

3.2. Conformal prediction

The high-level goal of conformal prediction (CP), first introduced in [12], is to associate an arbitrary confidence level with the output of an already-trained classifier. Specifically, given a target *coverage level* $1-\alpha \in [0,1]$ and a classifier, for a given input sample x CP will construct a random a *prediction set* $C(x)$ such that it is guaranteed to contain the correct class lies in with a probability close to $1-\alpha$. Intuitively, better classifiers and higher miscoverage levels will result in smaller prediction sets: a perfect classifier will have $|C| = 1$ for any miscoverage level. On the other hand, requiring a miscoverage level of $\alpha = 0$ with any non-ideal classifier will result in prediction sets including most if not all the possible classes.

CP constructs prediction sets through a *calibration set*, composed of n pairs of samples and labels (x_j, y_j). Then, during what is known as the *calibration* process, it computes a conformity score s_j for each sample x_j of the calibration set, expressing how well a candidate label conforms to the predictive distribution. Given these scores, the prediction set is then formed by including all labels whose test score is below a suitably chosen quantile of the calibration scores. The CP guarantee is then stated as:

$$1-\alpha \le \mathbb{P}\left(y_j \in C(x_j)\right) \le 1-\alpha+\frac{1}{n-1}.$$

As per the last term of the above equation, we can get arbitrarily close to the target miscoverage level by adding elements to the calibration set, i.e., increasing n.

The selection of the conformity score is itself an important decision, fraught with consequences. Existing approaches include adaptive prediction sets [13] (APS), where the conformity score ranks labels according to the predicted probability and includes them in decreasing order until the true label is covered. In general, there is a trade-off between the need for smaller coverage sets and the ability to account for how difficult to classify a certain instance is.

Regardless of the selected conformity score, the associated conformal sets tend to contain classes which may, with high probability, apply to the given sample. In other words, such sets contain *(i)* the correct classes, and *(ii)* classes which might be returned by the classifier in case of an incorrect decision. It follows that *classes jointly appearing in conformal sets can be thought of as easy to confuse*. We will exploit this intuition for our definition of CCMs, as set forth below.

4. The conformal correlation matrix

The high-level purpose of our Conformal Correlation Matrices (CCMs) is to convey information on pairs of classes that may be confused with one another. Importantly, we are not only interested in samples that *are misclassified*, but also *near misses*, i.e., situations where an incorrect classification decision is associated with a high probability (e.g., a high value of the corresponding logits). Notice that these situations need not result in low average classification accuracy. However, they may lead to low classification accuracies for classes that are considered to be particularly relevant or hard to identify, i.e., *critical classes*, or bring about a sudden, major performance drop in the case of changes in the input distribution.

Traditionally, pairs of easy-to-confuse classes are detected through confusion matrices (CM). However, it is worth remarking that CMs only detect *current* classification errors, not potential ones. In other words, they are useful tools to *fix* training issues with the current model. Instead, our goal is to detect and address such issues *proactively*, before the model has the ability to distinguish between critical classes.

To achieve this goal, we study the statistical dependencies among classes included in conformal prediction sets. The underlying idea is that correlations between classes which appear in these sets can reveal whether a model consistently confuses related categories (i.e., classes that are "easy to confuse" and/or are semantically related), or whether the inclusion of one class systematically excludes another (i.e., the classes that are "hard to confuse"). By capturing such relationships, we obtain a deeper, more nuanced view of the classifier reliability and predictive uncertainty, complementing and extending the insights offered by standard evaluation tools like confusion matrices.

To define CCMs, let H denote the number of possible classes and let us define the random indicator vector $\mathbf{z} = [z_1, \ldots, z_H]$ associated with a fresh sample x_{test}, where each component of the vector is defined as

$$z_h = \begin{cases} 1, & \text{if } h \in C(x_{\text{test}}), \\ 0, & \text{if } h \notin C(x_{\text{test}}), \end{cases}$$

where $C(x_{\text{test}})$ denotes the conformal prediction set produced by the classifier for sample x_{test}.

The covariance matrix $\mathbf{\Sigma}$ of $\mathbf{z}$ is defined entrywise as

$$\Sigma_{ij} = \mathbb{E}[z_i z_j] - \mathbb{E}[z_i]\mathbb{E}[z_j], \tag{1}$$

and the CCM $\mathbf{R}$ has entries ρ_{ij} given by

$$\rho_{ij} = \frac{\Sigma_{ij}}{\sqrt{\Sigma_{ii}\,\Sigma_{jj}}}. \tag{2}$$

We refer to the matrix $\mathbf{R}$ as the conformal correlation matrix (CCM). Its entries ρ_{ij}, which in the interval $[-1, 1]$, quantifies the correlation between the inclusion of classes i and j in a conformal prediction set, i.e., more intuitively, how easy to confuse those classes are.

We make the following remarks.

Remark 1. Intrinsically similar classes might be expected to exhibit high positive correlation in the CCM entries, that is, $\rho_{ij} > 0$, since a model that lacks sufficiently discriminative features is likely to include them together in the same prediction sets.

Remark 2. Pairs of dissimilar classes are expected to display strong negative correlation, corresponding to $\rho_{ij} < 0$, as the inclusion of one class in a prediction set typically reduces the likelihood of the other being included.

Remark 3. classes that are unrelated or independent with respect to the classifier representation are expected to yield correlation coefficients close to zero, i.e., $\rho_{ij} \approx 0$.

In practical settings, both centralized and distributed, CCMs can be estimated by first training the model, then computing conformity scores using a calibration set, and finally evaluating the conformal prediction sets on the test set. An important feature of the CCMs is that labels are not required to generate the conformal prediction sets used for estimating the correlation structure. It follows that *the CCMs can be estimated entirely from unlabeled samples*. This property is particularly valuable in scenarios where label information cannot be disclosed, for example in privacy-sensitive applications or distributed (including FL) settings.

We now discuss the concrete benefits of CCMs and how they help characterize and improve the training process, in both centralized (Section 5 next) and distributed (Section 6) settings.

5. The benefits of CCMs: Centralized setting

In this section, we discuss and detail the benefits of CCMs as a means to characterize and drive the improvement of a classifier's performance. Specifically, using CCMs allows a deeper view of classification performance, including potential classification errors and near misses, and it provides a reason for the classification errors that do happen.

We start by considering a centralized learning scenario and, for sake of concreteness, we demonstrate the benefits of CCMs with the help of a set of experiments performed with the CIFAR-100 and CIFAR-10 datasets. We will discuss how the benefits of CCMs extend to distributed settings, in Section 6.

Experimental settings

For our experiments, we consider an image classification task, leveraging the ResNet-18 model [14]. The model is trained for 50 epochs, using the stochastic gradient descent (SGD) optimizer with an initial learning rate of 0.01, momentum of 0.9, weight decay of 5×10^{-4}, and a step decay schedule where the learning rate is multiplied by $\gamma = 0.5$ every 10 epochs. Training takes places in batches, whose size is 64 samples. 80% of the samples of each dataset are used as training set, 10% as testing set, and 10% as calibration set which is used to compute the conformity scores. The miss-coverage level values considered are $\alpha \in \{0.3, 0.2, 0.1, 0.05\}$. Furthermore, we use the APS conformity scores [13], though regularized versions like RAPS [15] could be employed as well.

Benefit 1: A deeper view of classification performance

A major benefit of CCMs, especially when compared to traditional tools like confusion matrices, is that they offer more information about how the classifier operates, and how its performance can be improved. Indeed, as mentioned, confusion matrices can only convey information about *actual* classification errors. By contrast, CCMs also reflect *potential* errors, allowing us to address them in a proactive manner.

This is confirmed by the fact that, in scenarios with a large number of classes, it is not uncommon for certain entries of the confusion matrix to be vanishingly small. However, this does not necessarily indicate that the corresponding classes are dissimilar or that there is no risk of misclassification. As CCM entries incorporate the entire distribution of predicted probabilities, they are a much more effective tool to reveal subtle, potential classification issues.

To demonstrate this, we consider a subset of 10 classes taken from CIFAR-100, to wit, those belonging to the "aquatic mammals" and "fish" superclasses. We identify class 91 as a *critical class*, i.e., a class that is especially important to identify correctly. Fig. 1 reports the resulting confusion matrix, highlighting two examples of classes that have zero misclassification error with our critical class, i.e., 91 and 30. In other words, the confusion matrix tells us that the classifier never outputs class 30 *or* class 95 when the correct class is 91. However, this picture is not complete; as an example, in many cases, we could be interested in assessing *which*, between classes 30 and 95, is more likely to be confused with our focus class.

The answer comes from CCMs, reported in Fig. 2 for three values of α. We can see that the CCM entry associated with classes 91 and 95 is negative, while that between classes 91 and 30 is positive. This indicates that the classifier is much more likely to misclassify images of class 91 as class 30 than as class 95. Such information represents an additional insight into the DNN training status; even more importantly, it can drive and inform further refinement of the classifier. As an example, whenever new data can be added to the training set, additional images of class 30 would be preferable, so as to further reduce the likelihood of confusing classes 30 and 91.

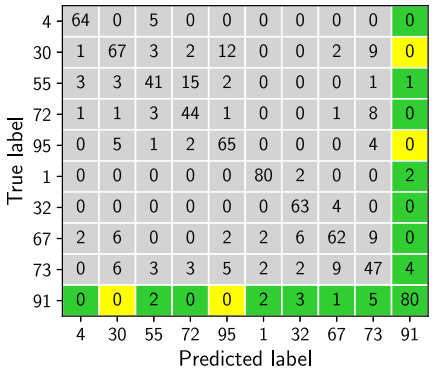


Fig. 1. CIFAR-100 experiments in a centralized setting: confusion matrix for the classes belonging to the "aquatic mammals" and "fish" superclasses. Yellow entries correspond to pairs of classes with zero misclassifications.

Benefit 2: Detecting the reason for classification errors

In many cases, we are interested in the underlying reason behind classification errors. In other words, we need to distinguish between classes that are confused with each other because they are intrinsically correlated (e.g., because the corresponding images do look similar) or simply because the classifier is in the early stages of training. To this end, we focus on the CIFAR-10 dataset and consider two pairs of classes as critical classes:

- "cat" and "dog", which are intrinsically similar;
- "car" and "plane", which a properly trained classifier should have no problem distinguishing.

We then compare the ability of CCM and confusion matrix entries to tell these two cases apart.

We begin with Fig. 3, showing the evolution of CCM entries for the "cat" and "dog" (left), and the "car" and "plane" (right) pairs of classes. We can immediately identify a different behavior. For the two intrinsically similar classes (left), the CCM entries start from higher values and steadily decrease with additional training. For the other classes, the CCM entries drop quickly after the first training epochs and remain steady for the remainder of training. The former behavior corresponds to a classifier that slowly learns to distinguish intrinsically similar classes, the latter to a classifier that quickly reaches the ability to separate distinct classes.

Looking at the corresponding values of the confusion matrix entries, reported in Fig. 4, we can observe a very similar behavior for the two pairs of classes. As discussed above, the confusion matrix tells us which classes are *currently* misclassified. In contrast, the CCM provide additional insights in the underlying reason for misclassifications – both actual and potential, as seen above –, hence, it is a precious tool to address and prevent such errors.

Benefit 3: Adaptability to training settings and phases

A third benefit of the CCMs lies in their parametric nature, i.e., in the fact that conformal sets, and therefore the CCMs themselves, are computed for a specific value of the miscoverage factor α. As discussed earlier, the CCMs contain information on potential misclassifications as well as actual ones. By tweaking the value of α, we can specify the likelihood of potential misclassifications between the classes we are interested in — or, equivalently, how "near" the "near misses" have to be.

This feature makes the CCM a tool suitable for different scenarios and conditions — most notably, α can be adjusted to reflect the severity of the consequences of classification errors. At the same time, the

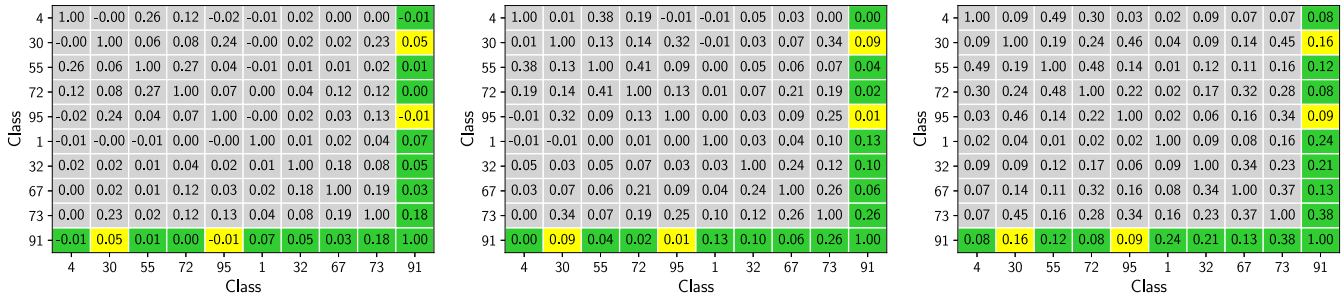


Fig. 2. CIFAR-100 experiments in a centralized setting: the matrix **R**, reporting CCM entries between all pairs of classes, when $\alpha = 0.3$ (left), $\alpha = 0.2$ (center), and $\alpha = 0.1$ (right). The highlighted entries correspond to the same pairs of classes highlighted in Fig. 2.

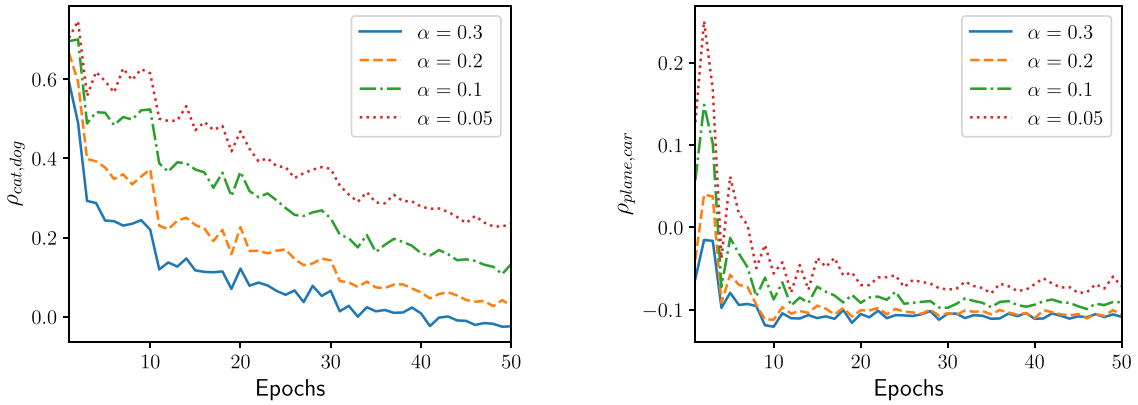


Fig. 3. CIFAR-10 experiments in a centralized setting: evolution of the CCM entry across training epochs for the "cat" and "dog" classes (left), and the "car" and "plane" classes (right).

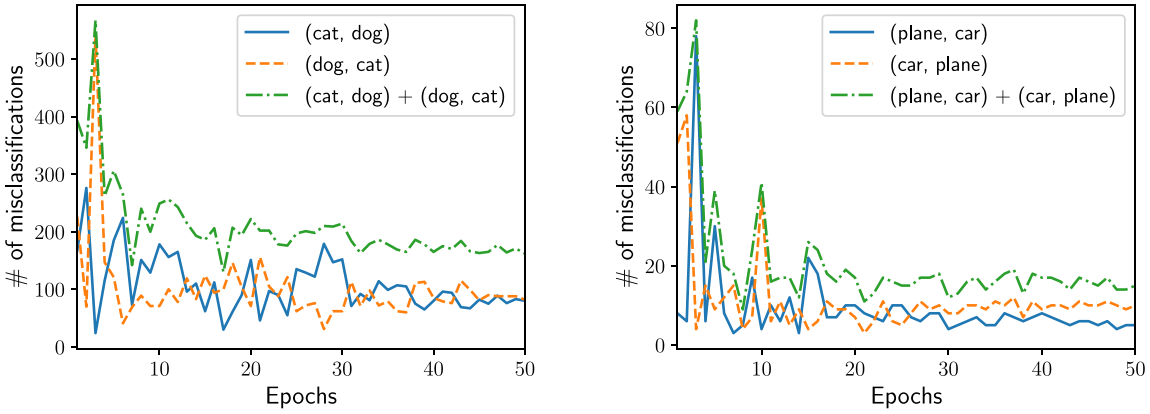


Fig. 4. CIFAR-10 experiments in a centralized setting: evolution of the confusion matrix entry across training epochs for the "cat" and "dog" classes (left), and the "car" and "plane" classes (right).

α parameter allows the CCMs to adapt to different training phases; as an example, we could decrease α as training progresses. This is especially useful when the pre-training and refinement stages are separate, hence, potentially, they have different objectives. Alternatively, α can be adjusted according to the application and the severity of the consequences of making errors.

6. Evaluating the benefits of CCMs in FL

In this section, we discuss and demonstrate the benefits brought by the CCMs in improving learning performance in a distributed setting and, specifically, when the FL paradigm is adopted. Specifically, CCMs preserve privacy to a better extent than traditional approaches like CMs, and they can be used for client selection. The experimental settings we used and the obtained results are presented below.

Experimental setting

We consider an FL scenario where a total of $N = 10$ clients train a ResNet-18 model [14] for image classification, over the CIFAR-10 and CIFAR-100 datasets. Each client has a different sample size and not all classes are represented equally in all clients' datasets – i.e., we are in a non-i.i.d. setting. Specifically, the number of samples of each class in each local dataset follows a Dirichlet distribution, with concentration parameter $\beta \in \{10.0, 1.0, 0.5\}$; higher values of β correspond to more heterogeneous local distributions. At each round, the FL server selects three clients to participate in the training, and combines local updates through the vanilla FedAvg algorithm.

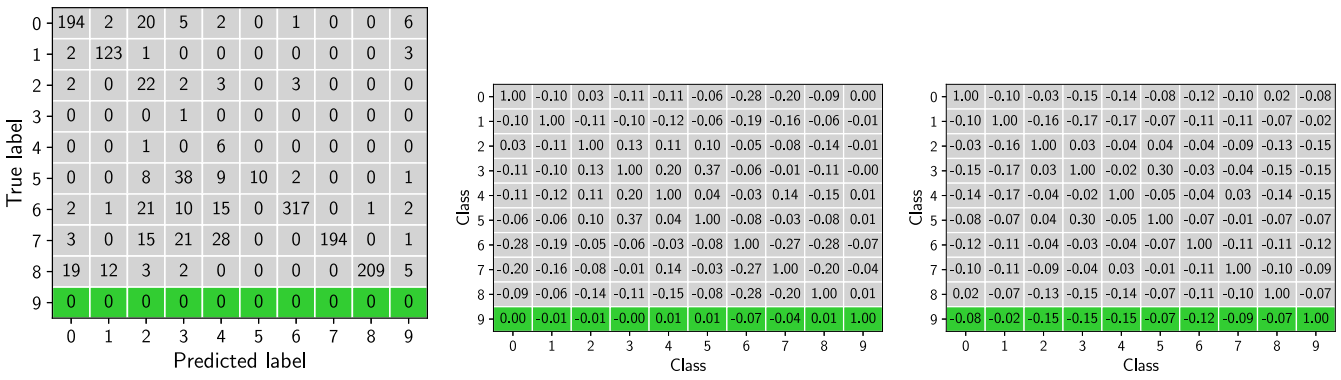


Fig. 5. CIFAR-10 experiments in an FL setting, $\alpha = 0.3$: confusion matrix for client 1 (left); CCM entries for client 1 (center); CCM entries computed with a global calibration set (right).

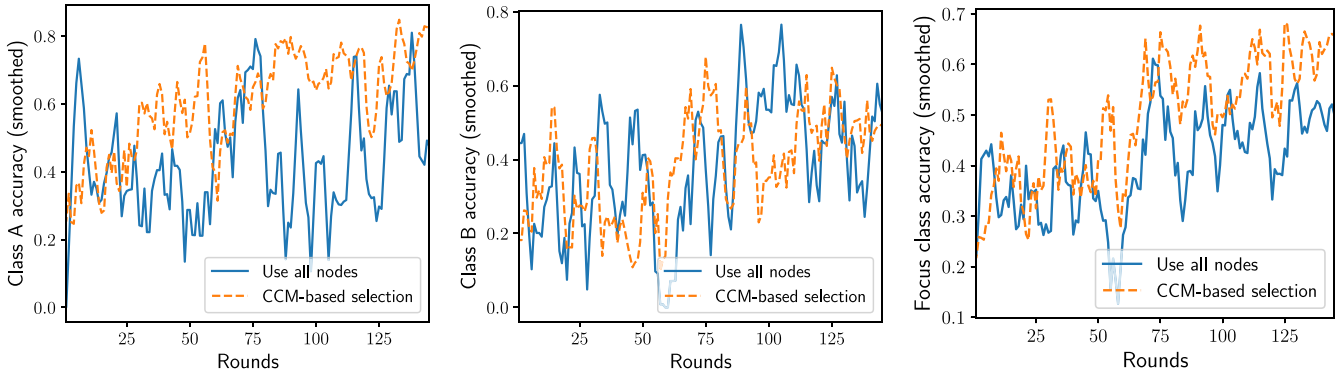


Fig. 6. CIFAR-10 experiments in an FL setting: CCM entries for the critical classes under the "all" and "exclude low CCM entry" client selection strategies, when $\alpha = 0.3$.

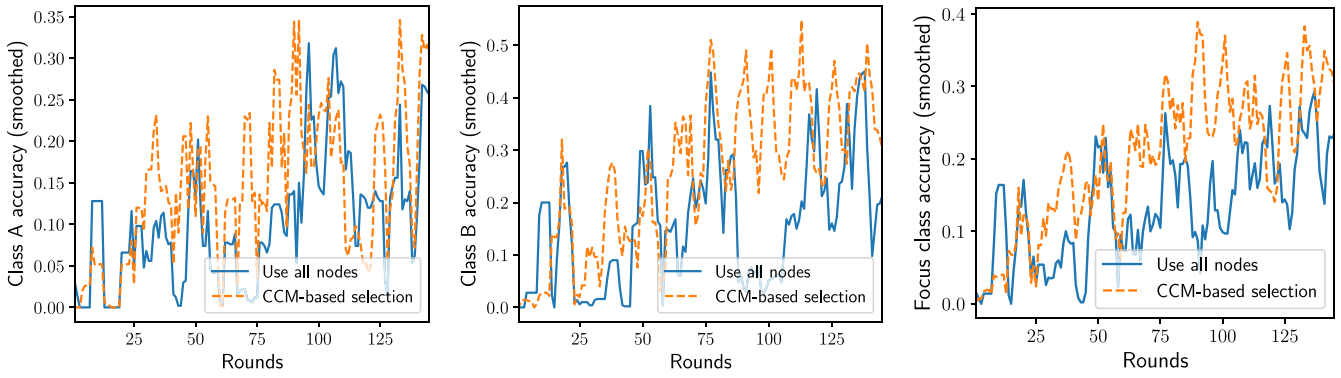


Fig. 7. CIFAR-100 experiments in an FL setting: CCM entries for the critical classes under the "all" and "exclude low CCM entries" client selection strategies, for $\alpha = 0.3$.

Table 1
Summary table: Performance of the considered selection strategies.

Dataset	Strategy	# Selected	Accuracy				
			Overall	Focus classes	FC A	FC B	Other classes
CIFAR-10	Use all clients	10	0.789	0.510	0.808	0.211	0.859
	CCM-based	6	0.801	0.583	0.742	0.423	0.856
CIFAR-100	Use all clients	10	0.511	0.415	0.630	0.200	0.513
	CCM-based	7	0.493	0.450	0.570	0.330	0.494

Benefit 4: Privacy

As a part of the FL process, clients share information with the FL server. On the one hand, such information is necessary for the server to make effective decisions, e.g., selecting the best clients and properly weighting their updates. On the other hand, any information disclosed by clients is a potential privacy breach. An especially important piece of information needed by the server is how well the global model is performing at each of the clients, i.e., over the local datasets; however, such datasets cannot, in general, be shared.

A way to convey this information through confusion matrices is having clients send their CMs to the server, as exemplified in Fig. 5(left). This allows the server to know which clients are struggling, and which classes are problematic for each client. On the negative side, by sharing its confusion matrix, each client also discloses *(i)* how large its local dataset is, and *(ii)* how many samples of each class it contains indeed, both can be obtained by simply summing rows in the CM. Depending upon the concrete scenario and conditions, sharing such information could create privacy – and, possibly, even legal – issues.

As an alternative, clients can share their correlation matrix, exemplified in Fig. 5(center). As discussed earlier, this matrix contains a superset of the information conveyed by the confusion matrix; therefore, it can be used by the server *in lieu* of the confusion matrix. At the same time, the correlation matrix only contains *probabilities*; hence, it does not leak information on the size and composition of the local datasets. In summary, CCMs can be exchanged as part of the FL process to effectively manage the process itself, preserving the privacy of clients, and without adverse effects on learning performance.

It is also interesting to compare the local CCMs in Fig. 5(center) to the global ones in Fig. 5(right). We can see that, while the numerical values of the entries change, the patterns remain unchanged. In other words, it is possible to obtain a good picture of how the overall training proceeds without sharing local data or local information between clients.

Benefit 5: Client selection

Client selection is arguably one of the most consequential decisions in optimizing FL training. Broadly speaking, the FL server has to select the clients that maximize the quality of learning, while accounting for the fact that additional clients may slow down individual FL rounds and/or increase learning costs. Traditional ways of assessing how useful a client can be to FL include considering the size [16] and diversity [17] of local datasets, with more recent approaches also accounting for local resources [18] and connectivity [19]. However, when the ability of the classifier to distinguish critical classes is the main learning quality metric, such approaches may yield suboptimal decisions.

We shall turn to the CCMs as a means to drive higher-quality decisions. Specifically, we identify two classes for each dataset as critical; hence, our goal is to make the corresponding CCMs to be as high as possible. We consider two client selection strategies:

- "Use all nodes": selecting, at each round, three out of the 10 clients (same as for Fig. 5 earlier);
- "CCM-based selection": selecting, at each round, three out of the 7 clients with the highest CCM entries for the critical classes.

The intuition behind the latter strategy is that, if a client is unable to distinguish the critical classes on its local dataset, its local updates are unlikely to help the convergence of the global classifier model.

Fig. 6 shows the evolution of the CCM entries between the critical classes across rounds, with each line corresponding to a client selection strategy. We can observe that, for all values of α, excluding the clients with the smallest *local* CCMs results in a significant improvement in *global* CCM entries. The trend remains unchanged if we extend the experiments to CIFAR-100, as reported in Fig. 7, and highlights how the insights provided by the CCMs can be acted upon by FL servers to make their decisions, including client selection.

Finally, Table 1 shows the accuracy achieved by all classes, for, respectively, the CIFAR-10 and CIFAR-100 datasets. We can observe that using CCMs to make node selection decisions results in:

- a significant improvement in the accuracy of the critical class with the lowest accuracy;
- a slight decrease in the accuracy of the critical class with the highest accuracy;
- small effects on the accuracy of non-critical classes;
- virtually no effect on the average accuracy.

We can thus conclude how using the CCMs can improve the classifier's ability to distinguish critical classes without hurting the overall classification performance; specifically, it does not affect the performance for non-critical classes.

7. Conclusions

In this work, we have introduced the Conformal Correlation Matrix (CCM), a novel metric designed to complement traditional evaluation tools for classification models. Unlike the confusion matrix, which reflects actual misclassifications, the CCM captures the likelihood of potential ambiguities between classes based on conformal prediction outcomes. This allows for a deeper understanding of model behavior, providing actionable insights into how and why certain misclassifications may arise. Through experiments on the CIFAR-10 and CIFAR-100 datasets, we have shown that CCMs can reveal subtle relationships between classes that are not visible through standard accuracy or confusion-based analyses. By doing so, CCMs enable more informed refinement of training strategies, data augmentation, and model tuning. We have further demonstrated the applicability of CCMs in federated learning environments, where data heterogeneity, privacy constraints, and communication costs complicate the training process.

Our results show that CCMs can be leveraged both as privacy-preserving performance indicators and as effective tools for guiding client selection. In particular, selecting clients with higher CCM values for critical classes leads to improved discrimination performance without compromising overall accuracy. This highlights the potential of CCM-based methods to balance utility and privacy in distributed learning scenarios.

Future work will extend this framework by integrating CCM-driven feedback directly into the model optimization loop and exploring adaptive strategies for selecting the training process parameters as well as the clients to be involved in the process. We believe that the proposed approach opens promising directions for trustworthy, interpretable, and privacy-preserving learning in both centralized and federated settings.

CRediT authorship contribution statement

Alessandro Perlo: Writing – original draft, Validation, Software, Methodology, Investigation, Formal analysis, Data curation. **Carla Fabiana Chiasserini:** Writing – review & editing, Supervision, Resources, Methodology, Conceptualization. **Gustavo De Veciana:** Writing – review & editing, Supervision, Methodology, Formal analysis, Conceptualization. **Francesco Malandrino:** Writing – original draft, Validation, Software.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

[1] D. Stutz, Krishnamurthy, Dvijotham, A.T. Cemgil, A. Doucet, Learning optimal conformal classifiers, in: ICLR, 2022.

[2] C. Lu, Y. Yu, S.P. Karimireddy, M. Jordan, R. Raskar, Federated conformal predictors for distributed uncertainty quantification, in: International Conference on Machine Learning, PMLR, 2023, pp. 22942–22964.

[3] V. Plassier, M. Makni, A. Rubashevskii, E. Moulines, M. Panov, Conformal prediction for federated uncertainty quantification under label shift, in: International Conference on Machine Learning, PMLR, 2023, pp. 27907–27947.

[4] P. Humbert, B. Le Bars, A. Bellet, S. Arlot, One-shot federated conformal prediction, in: A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, J. Scarlett (Eds.), Proceedings of the 40th International Conference on Machine Learning, in: Proceedings of Machine Learning Research, vol. 202, PMLR, 2023, pp. 14153–14177, URL https://proceedings.mlr.press/v202/humbert23a.html.

[5] H. Hsu, H. Qi, M. Brown, Measuring the effects of non-identical data distribution for federated visual classification, 2019, URL https://arxiv.org/abs/1909.06335.

[6] F. Lai, X. Zhu, H.V. Madhyastha, M. Chowdhury, Oort: Efficient federated learning via guided participant selection, in: 15th USENIX Symposium on Operating Systems Design and Implementation, OSDI 21, USENIX Association, 2021, pp. 19–35, URL https://www.usenix.org/conference/osdi21/presentation/lai.

[7] Y. Jee Cho, J. Wang, G. Joshi, Towards understanding biased client selection in federated learning, in: G. Camps-Valls, F.J.R. Ruiz, I. Valera (Eds.), Proceedings of the 25th International Conference on Artificial Intelligence and Statistics, in: Proceedings of Machine Learning Research, vol. 151, PMLR, 2022, pp. 10351–10375, URL https://proceedings.mlr.press/v151/jee-cho22a.html.

[8] W. Chen, S. Horváth, P. Richtárik, Optimal client sampling for federated learning, Trans. Mach. Learn. Res. (2022) URL https://openreview.net/forum?id=8GvRCWKHIL.

[9] H. Wu, P. Wang, Node selection toward faster convergence for federated learning on non-IID data, IEEE Trans. Netw. Sci. Eng. 9 (5) (2022) 3099–3111, http://dx.doi.org/10.1109/TNSE.2022.3146399.

[10] A. Negrão, G. Silva, R. Pedrosa, E. Luz, P. Silva, Adaptive client-dropping in federated learning: Preserving data integrity in medical domains, in: Intelligent Systems: 34th Brazilian Conference, BRACIS 2024, BelÉm Do Pará, Brazil, November 17–21, 2024, Proceedings, Part I, Springer-Verlag, Berlin, Heidelberg, 2025, pp. 111–126.

[11] J. Konečný, B. McMahan, D. Ramage, Federated optimization: Distributed optimization beyond the datacenter, 2015, arXiv preprint arXiv:1511.03575.

[12] A.N. Angelopoulos, S. Bates, A gentle introduction to conformal prediction and distribution-free uncertainty quantification, 2021, URL https://arxiv.org/abs/2107.07511.

[13] Y. Romano, M. Sesia, E. Candes, Classification with valid and adaptive coverage, in: H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, H. Lin (Eds.), Advances in Neural Information Processing Systems, vol. 33, Curran Associates, Inc, 2020, pp. 3581–3591.

[14] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[15] A.N. Angelopoulos, S. Bates, J. Malik, M.I. Jordan, Uncertainty sets for image classifiers using conformal prediction, 2020, arXiv preprint arXiv:2009.14193.

[16] M. Kamp, J. Fischer, J. Vreeken, Federated learning from small datasets, in: ICLR, 2023.

[17] Z. Li, Y. He, H. Yu, J. Kang, X. Li, Z. Xu, D. Niyato, Data heterogeneity-robust federated learning via group client selection in industrial IoT, IEEE Internet Things J. 9 (2022).

[18] A. Imteaj, M.H. Amini, Fedar: Activity and resource-aware federated learning model for distributed mobile robots, in: IEEE ICMLA, 2020.

[19] Y. Zhou, Q. Ye, J.C. Lv, Communication-Efficient Federated Learning with Compensated Overlap-FedAvg, IEEE Trans. Parallel Distrib. Syst. (2021).