Contents lists available at ScienceDirect

Biomedical Signal Processing and Control

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

Uncertainty estimation of deep learning models for atrial fibrillation detection from Holter recordings: A benchmark study

Md Moklesur Rahman [a,*], Massimo Walter Rivolta [a], Fabio Badilini [b,c], Roberto Sassi [a]

[a] Dipartimento di Informatica, Università degli Studi di Milano, Via Celoria 18, Milan, 20133 MI, Italy
[b] Center for Biosignal Research, Department of Cardiology, University of California, San Francisco, 94143 CA, USA
[c] AMPS-LLC, New York, 10025 NY, USA

ARTICLE INFO

Keywords:
Uncertainty quantification
Deep learning
Atrial fibrillation
Bayesian deep learning

ABSTRACT

With the development of deep learning (DL)-based methods, automated atrial fibrillation (AF) detection from electrocardiograms (ECGs) has recently gained much attention. Although the performance of DL has been encouraging, the susceptibility of DL models to overfitting would benefit from the exploration of uncertainty quantification (UQ) to ensure safe integration into clinical practice. However, there has been limited exploration of UQ methods in the context of DL models for AF detection using Holter ECG recordings, and a comprehensive comparison of various UQ techniques remains absent. This study addressed this gap by introducing a benchmark study wherein 11 distinct UQ methods were rigorously evaluated and compared across three public Holter repositories: IRIDIA-AF, Long-Term AF, and MIT-BIH AF datasets. A residual DL model was used for the UQ methods, which is one of the most common architectures in this domain for its ability to capture complex patterns within ECG data. The findings revealed that batch-ensemble (BE) and packed-ensemble (PE) outperformed other UQ methods concerning both performance, as quantified by sensitivity, specificity and expected calibration error, and computational efficiency. In addition, when we implemented reject inference to discard ECG segments where the model confidence was not sufficiently high, BE and PE still showed to reject the least number of samples, while retaining the highest detection performance.

1. Introduction

Atrial fibrillation (AF), a cardiac arrhythmia characterized by irregular electrical activity in the atria, poses a substantial risk of stroke and other cardiovascular complications [1]. Early detection of AF is paramount, as it enables timely implementation of preventive measures, thereby reducing the risk of stroke and other adverse cardiovascular outcomes. Traditionally, diagnostic electrocardiograms (ECG) have been used as a gold-standard technique for AF diagnosis. However, challenges arise, especially in identifying non-persistent AF episodes during routine clinic monitoring. In response to these challenges, Holter monitoring is performed for extended periods (24, 48 h or more) [2], also during follow-up after catheter ablation [3]. As continuous monitoring yields copious ECG data, manual beat-by-beat analysis becomes impracticable. Consequently, the development of reliable automated systems becomes imperative to aid cardiologists in efficiently identifying AF episodes.

With the rapid evolution of artificial intelligence (AI), machine learning (ML) models have emerged as promising tools to enhance AF detection accuracy [4]. Nevertheless, the development of ML techniques for AF detection requires manual feature extraction and domain expertise. This process is further complicated by noise and variability in the data, hindering robust feature extraction. Therefore, innovative approaches are needed to overcome these challenges and fully harness the potential of ML in improving AF detection accuracy.

In recent years, deep learning (DL) methods, particularly convolutional neural networks (CNNs), have demonstrated promising results in detecting AF [4–7]. Despite achieving performance levels comparable to those of cardiologists, concerns remain regarding the reliability and acceptance of DL models in clinical settings. Variability in ECG signal characteristics, including artifacts and noise, as well as the diversity of ECG signals beyond the training data, contribute to these concerns. Additionally, DL models may exhibit inconsistent performance on new data, which undermines medical professionals' trust in their integration into clinical practice.

To address these issues, it is crucial to provide clinicians with supplementary information, such as the confidence levels associated

* Corresponding author.
E-mail addresses: md.rahman@unimi.it (M.M. Rahman), massimo.rivolta@unimi.it (M.W. Rivolta), badilini@amps-llc.com (F. Badilini), roberto.sassi@unimi.it (R. Sassi).

https://doi.org/10.1016/j.bspc.2025.109032
Received 25 February 2025; Received in revised form 8 October 2025; Accepted 31 October 2025
1746-8094/© 2025 Published by Elsevier Ltd.

with model outputs, rather than merely presenting the outputs themselves. The increasing volume of ECG recordings requiring interpretation exacerbates the need for efficient review processes, as manual review of automated ECG analyses becomes progressively time-consuming and resource-intensive. By providing uncertainty information about model outputs, clinicians can prioritize cases where the models exhibit uncertainty, thereby optimizing the allocation of time and resources.

Uncertainty quantification (UQ) can be a possible direction to alleviate the reliability problem of DL models. Uncertainty in DL models reflects the confidence levels associated with predicting ECG rhythms, distinguishing, for example, between AF and Non-AF rhythms. Two distinct types of uncertainty can manifest in DL classifiers: data (aleatoric) uncertainty and model (epistemic) uncertainty [8]. Data uncertainty arises from various sources such as noisy sensors, errors in data collection, and ambiguity in data labeling. In contrast, model uncertainty stems from a lack of knowledge regarding the model parameters, particularly evident when the model is trained on limited or insufficient data, resulting in gaps in its representation of underlying data patterns. Given these potential sources of uncertainty, integrating UQ techniques is necessary to develop robust and reliable DL models for AF detection. From a clinical standpoint, uncertainty estimates offer valuable insights to guide or automate labeling corrections, reject DL outputs with insufficient certainty, and aid in detecting classification failures at the patient level.

Various methods for UQ in DL models are available [9–12], some of which have been employed in the context of AF detection [13–20]. For instance, variational methods represent capturing uncertainty by considering the weights of a DL model as random variables, and approximating their joint (posterior) distribution through variational inference (VI) [13,15,21]. However, VI-based methods pose limitations in implementation and training, leading to scalability issues in both architecture and data size. Ensemble methods are another popular method for UQ, where multiple models are trained from scratch, and their performance is computed based on average predictions [15,22]. In the existing literature, the predominant focus has been on investigating deep ensemble (DE) [15] and Monte Carlo dropout (MCD) [16] methods. However, a notable limitation of these approaches is their lack of scalability [23]. Although innovative UQ methods exist across various domains, their application for AF detection has not been widely explored in prior research. Therefore, there is a critical need for a thorough examination of diverse UQ methods specifically tailored for AF detection across various datasets. This study addresses this gap by conducting experiments on three public datasets, considering both internal and external validation sets, to compare the performance and identify the most suitable UQ method for AF detection. The main contributions of this study are as follows:

- We investigated 11 different UQ methods tailored for detecting AF using Holter recording data.
- We thoroughly evaluated these UQ techniques by adding random noise to the data, effectively simulating real-world scenarios.
- We analyze the performance of the UQ methods across different rejection thresholds, providing valuable insights into their robustness and reliability in AF detection.

2. Related works

In recent years, interest has grown in understanding how uncertainty is managed in detecting AF using DL models.

Belen et al. [13] employed a variational autoencoder DL model, integrating the Kullback–Leibler (KL) Divergence loss function, for AF detection using the MIT-BIH atrial fibrillation (MIT-BIH-AF) dataset. To assess uncertainty, they iteratively fed the input data through the DL model and computed the standard deviation of the softmax probabilities. Vranken et al. [15] explored several UQ methods *e.g.*, MCD, VI, DE, and snapshot ensemble (SE) techniques. The efficacy of these methods in estimating uncertainties was assessed using rank-based metrics, calibration assessment, and out-of-distribution (OOD) detection. The findings revealed that VI with Bayesian decomposition and ensemble methods with auxiliary output exhibited superior performance.

In [16], a weakly supervised learning approach was developed by incorporating the MCD approach to consider a limited amount of labeled data. The model achieved a classification performance with an F1-score ranging from 0.64 to 0.67 and an expected calibration error (ECE) ranging from 0.05 to 0.07. Aseeri et al. [14] developed a gated recurrent unit-based DL model trained using three types of datasets and estimated uncertainty using MCD and DE methods. They demonstrated that DE methods outperformed the MCD method. Elul et al. [17] conducted an extensive investigation into the integration of AI within clinical settings, focusing on the crucial role of uncertainty estimation in managing OOD instances and enabling multilabel diagnoses. Their approach involved the development of a DL model comprising 10 binary classifiers, each corresponding to distinct trained ECG abnormalities. This design facilitated the model's capacity to identify any combination of recognized rhythms and to address unknown classes when the model generated negative predictions across all binary classifications. To gauge prediction confidence, they implemented the MCD method.

Zhang et al. [24] employed a Bayesian DL model with MCD for arrhythmia classification with a rejection option. They computed total uncertainty using an entropy-based decomposition of data and model uncertainty, and explored different uncertainty thresholds to improve classification performance by rejecting high-uncertainty instances. Jahmunah et al. [19] developed a Dirichlet distribution-based Densenet model with reverse KL divergence to compute predictive entropy for model uncertainty in a multi-class classification task. The authors argued that their approach was faster and computationally lightweight compared to previous uncertainty quantification methods. Additionally, they included noisy ECG in their analysis. Recently, Park et al. [18] proposed a self-attention-based LSTM-FCN DL architecture using a DE approach to quantify uncertainty. Their results achieved state-of-the-art performance, showing that epistemic uncertainty is reliable for classifying the six arrhythmia types they considered.

3. Dataset

In this study, three public ECG datasets are used to create the UQ benchmark: Long-Term atrial fibrillation (LTAF) dataset [25], IRIDIA-AF [26] and MIT-BIH-AF dataset [27]. The MIT-BIH-AF dataset is exclusively employed for testing purposes. Detailed descriptions of the three datasets are provided below:

- **LTAF**: This database contains 2-lead ECG signals from 84 patients of subjects with paroxysmal or sustained AF events with varying record durations but are typically 24 to 25 h [27]. The records are sampled at a frequency of 128 Hz. The rhythm annotations within the LTAF dataset are classified into two types: AF and N.
- **IRIDIA-AF:** This dataset comprises 167 Holter records from 152 patients experiencing paroxysmal AF, collected between 2006 and 2017 at an outpatient cardiology clinic in Belgium [26]. Each record provides a detailed snapshot of the patient's cardiac activity, meticulously annotated for AF episodes by both an expert cardiologist and a specialized cardiac nurse. The duration of these records varies, ranging from 19 h to a maximum of 95 h, with each segment divided into 24-hour files for ease of analysis. Holter recordings were sampled at a rate of 200 Hz.
- **MIT-BIH-AF**: This database contains 2-lead ECG signals from 23 patients sampled at a frequency of 250 Hz [25]. The rhythm types within MIT-BIH-AF are classified into four types: AF, AFL (atrial flutter), J (atrioventricular junctional rhythm), and N (sinus rhythm). In this study, the annotations of N are considered as "non-AF", while AF and AFL were merged as "AF".

Table 1
Number of 10-second ECG segments of different datasets.

Dataset	Training		Validation		Test	
	Non-AF	AF	Non-AF	AF	Non-AF	AF
LTAF	193,470	168,707	34,410	18,291	14,991	26,839
IRIDIA-AF	1,150,662	353,063	331,214	94,759	390,131	88,521
MIT-BIH-AF	–	–	–	–	42,041	26,247

3.1. Preprocessing

A third-order zero-phase Butterworth bandpass filter, with cut-off frequencies set at 0.5 Hz and 40 Hz, is employed to mitigate baseline wandering and powerline interference in the recordings. To ensure robust model development, validation, and testing, a patient-wise partitioning technique is implemented. The datasets are divided into training, validation, and testing sets in an 8:1:1 ratio, respectively. Each recording is subsequently segmented using non-overlapping 10-second windows. Table 1 provides a summary of the total number of AF and non-AF segments, each with a duration of 10 s, across the three datasets.

4. Model architecture

We consider a DL model whose architecture consists of 18 layers. To manage the optimization of such a complex network, shortcut connections are incorporated, similarly to a residual network architecture. The network comprises 8 residual blocks, each containing two convolutional layers. The number of residual blocks was selected by maximizing the accuracy on the validation set. These convolutional layers have a filter size of 3 and 32×2^k elements, where k is a hyper-parameter that starts at 0 and increments by 1 every two residual blocks. Additionally, every alternate residual block reduces the input size by a factor of 2 through subsampling.

To improve convergence and training stability, ReLU activation function and batch normalization are applied after each convolutional layer. Furthermore, dropout with a probability of 0.3 is introduced to prevent overfitting. Subsequently, two dense layers comprising 128 and 64 neurons are employed. Each dense layer is followed by ReLU activation, batch normalization, and a dropout layer. Ultimately, a softmax activation function is utilized to generate a probability in AF detection. The model architecture is depicted in Fig. 1. It is important to note that all UQ methods are employed within the same DL architecture. This architecture is inspired by previous studies that have applied a similar network structure for AF detection [21]. To accommodate differences in sampling rates, training is conducted on two datasets with distinct input tensor shapes: 1280 × 2 and 2000 × 2. Each tensor represents a 10-second, two-channel signal segment. The 1280 × 2 input corresponds to signals sampled at 128 Hz in the LTAF dataset, whereas the 2000 × 2 input corresponds to signals sampled at 200 Hz in the IRIDIA-AF dataset.

5. Uncertainty quantification methods

Bayesian inference diverges from deterministic predictions by embracing a probabilistic approach. Instead of providing a single, definitive answer, it considers a range of possible values for model parameters, facilitating the incorporation of prior knowledge and the refinement of beliefs based on observed data.

To formally illustrate this concept, let us consider a training dataset $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$ comprising N instances and labels, considered sampled from the random variable $(X, Y) \sim P_{X,Y}$. For simplicity, let $\mathbf{x}_i \in \mathbb{R}^d$ denote a vector and y_i a categorical variable. The input data $\mathbf{x}_i$ is fed into a neural network (NN) $\hat{y} = f_\theta(\mathbf{x}_i)$ with parameters θ, yielding a classification output. This NN is conceived as a probabilistic model, where $f_\theta(\mathbf{x}_i) = P(Y \mid X = \mathbf{x}_i, \theta)$ and, differently from a deterministic approach, θ is considered a random variable as well. The posterior distribution of θ given the observed training set $\mathcal{D}$ can be used as a proxy for UQ using Bayesian inference.

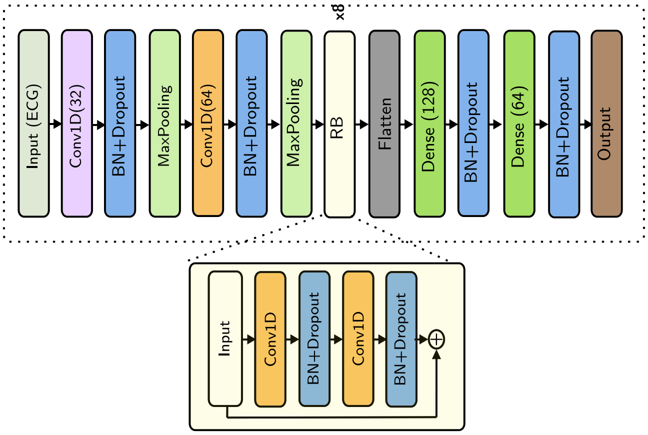


Fig. 1. Diagram of the DL model. BN and RB stand for batch normalization and residual block, respectively.

In the context of Bayesian modeling, ensemble methods provide a means to quantify uncertainty by combining multiple models. The parameters of each model within the ensemble represents a distinct sample of the posterior distribution over the model parameters. Having at disposal different models, the ensemble prediction $\hat{y}^{\text{ensemble}}$ is obtained by aggregating individual predictions from multiple models:

$$\hat{y}^{\text{ensemble}} = \frac{1}{M} \sum_{j=1}^{M} f_{\theta_j}(\mathbf{x}_i), \tag{1}$$

where $j = 1, 2, \ldots, M$ denotes the distinct models in the ensemble and $\hat{y}^{\text{ensemble}}$ indicates the probabilities for the label to predict. For all methods investigated in our study, we employ an ensemble of $M = 4$ across all UQ methods, each parameterized differently to capture a diverse set of hypotheses from the DL model. The details of the UQ methods are provided in the following subsections.

5.1. Monte Carlo dropout

MCD extends the traditional dropout regularization method [28]. In standard dropout, random units are dropped during training to prevent overfitting and encourage model robustness. MCD takes this concept further by employing dropout not only during training but also during the inference phase of a model. Instead of obtaining a single deterministic prediction, the model is run $M = 4$ times with dropout enabled, generating a distribution of predictions. The final prediction is then derived from the mean of these sampled predictions. In this study, we used a dropout rate of 0.3.

5.2. Ensemble method with different initializations

In this study, we leveraged the fact that the typical strategy for training a NN is to initialize its weights randomly and then adjusting them through back-propagation. Here, we trained $M = 4$ different NNs with four different initializations and obtained $\hat{y}^{\text{ensemble}}$ as the average of these four output probabilities [29].

5.3. Snapshot ensemble

SE creates multiple models through the training of a DL model using distinct snapshots of its parameters, obtained at various epochs during the training process [22]. These individual snapshots encapsulate the configuration of the model at different epochs, thereby offering diverse vantage points on the data manifold. The predictions derived from these varied models within the SE framework possibly serve not only to enhance predictive accuracy but also to furnish a more robust estimation of uncertainty belonging to the predictions of the models. In this study, with $M = 4$, we took a snapshot at every 20 epochs to develop the SE model.

5.4. Batch-ensemble

Unlike traditional ensembles that combine predictions from independently trained models, batch-ensemble (BE) utilizes ensemble members that share the same weights during training [9]. BE builds up an ensemble from a single base network (shared among ensemble members) and a set of layer-specific weight matrices unique to each member.

At each layer, the weight of each ensemble member is generated from the Hadamard product between a weight matrix shared among all ensemble members, called “slow weights” and a rank-one matrix that varies among all members, called “fast weights”. Formally, let $W_{\text{share}} \in \mathbb{R}^{u \times v}$ be the slow weights in an NN layer with input dimension u and output dimension v. Each member m from an ensemble of size M owns a fast weight matrix $W_m \in \mathbb{R}^{u \times v}$. W_m is a rank-one matrix computed from a tuple of trainable vectors $r_m \in \mathbb{R}^u$ and $s_m \in \mathbb{R}^v$, with $W_m = r_m s_m^\top$. BE generates from them the family of ensemble weights $\overline{W_m} = W_{\text{share}} \odot W_m$, where $\odot$ denotes the Hadamard product. Each member of the ensemble $\overline{W_m}$ is essentially a rank-one perturbation of the shared weights W_{share}. We implemented BE on all convolutional and dense layers in the NN. The loss function was binary cross-entropy averaged across the ensemble members.

5.5. Packed-ensemble

The utilization of ensemble methods is widely recognized for its advantages. However, a significant drawback is the considerable increase in both training time and memory usage during inference, which scales linearly with the number of models employed. To address these challenges, Olivier et al. [10] introduced the packed-ensembled (PE) method. This approach leverages grouped convolutions to significantly expedite the training and inference computations of ensembles. Grouped convolutions offer computational advantages by reducing the size of the subnetworks. Group convolutions can be extended to dense layers as well. Here, PE was used for all convolutional and dense layers and the number of groups was set to $M = 4$.

5.6. Mean field variational inference

Mean field variational inference (MFVI) is a technique used in the Bayesian framework to approximate complex posterior distributions [30]. The goal of MFVI is to approximate the true posterior distribution by parameterizing it with a simpler, factorized distribution, known as the mean field distribution. The true posterior is often difficult to compute analytically due to its complexity. MFVI seeks to approximate this distribution by a factorized parametric distribution that factorizes over the individual parameters:

$$q(\theta; \omega_1, \dots, \omega_K) = \prod_{i=1}^{K} q_i(\theta_i; \omega_i), \tag{2}$$

where θ_i represents the i-the parameter of the model, ω_i the parameters of the ith $q_i(\theta_i; \omega_i)$ distribution, $q(\theta; \omega_1, \dots, \omega_K)$ represents the complete variational distribution, and K is the total number of parameters. Each distribution $q_i(\theta_i; \omega_i)$ was taken to be a normal distribution over the variable θ_i. The mean field approximation implies that the parameters are assumed to be independent given the mean field distribution. The objective is to find the mean field parameters ω_i that minimize the Kullback–Leibler (KL) divergence between the true posterior and the mean-field approximation. Minimizing this divergence is equivalent to maximizing the Evidence Lower Bound (ELBO), which is defined as:

$$\text{ELBO} = \mathbb{E}_{q(\theta; \omega_1, \dots, \omega_K)}[\log p(Y|X, \theta) - \log q(\theta; \omega_1, \dots, \omega_K)] \tag{3}$$

which is a tractable objective function that can be optimized using various optimization algorithms, such as stochastic gradient descent (SGD). MFVI was implemented in the first and last layers of our NN.

5.7. Rank-one MFVI

The rank-one MFVI method aims to approximate complex probability distributions by introducing a simplified, tractable family of distributions [11]. This approach merged the key ideas from BE and MFVI by constructing a posterior distribution over the parameters of the rank-one matrices $rs^\top$. Similar to MFVI, we used the normal distribution for each of these parameters and the variational distribution was optimized using ELBO. Please note that, in this case, there are no four members, but only one. Rank-one MFVI was used for all convolutional and dense layers.

5.8. Stochastic weighting average Gaussian

Stochastic weighted average (SWA) centers around a learning rate schedule within SGD, and considers the weights of the models it encounters at consecutive epochs [31]. In this method, the weights obtained after each epoch, denoted as $\theta^{(e)}$, contribute to a running average, *i.e.*, the SWA solution, after T epochs: $\theta_{\text{SWA}} = \frac{1}{T}\sum_{e=1}^{T} \theta^{(e)}$.

Maddox et al. [12] extends this method to estimate Gaussian posteriors for model parameters, by also estimating a covariance matrix for the parameters, using a low-rank plus diagonal posterior approximation. The diagonal part is obtained by keeping a running average of the second uncentered moment of each parameter, and then at the end of the training calculating:

$$\Sigma_{\text{diag}} = \text{diag}\left(\frac{1}{T}\sum_{e=1}^{T} \theta^{(e)^2} - \theta_{\text{SWA}}^2\right), \tag{4}$$

while the diagonal part is approximated by keeping a matrix $GG^\top$ with columns $G_e = (\theta^{(e)} - \hat{\theta}^{(e)})$, $\hat{\theta}^{(e)}$ standing for the running estimate of the parameters' mean obtained from the first e epochs. The rank of the approximation is restricted by retaining last L vectors of the G_e vectors and dropping the previous, with L being a hyperparameter of the model, as follows

$$\begin{aligned}\Sigma_{\text{low-rank}} &\approx \frac{1}{L-1} GG^\top \\ &= \frac{1}{L-1} \sum_{e=T-L+1}^{T} (\theta^{(e)} - \hat{\theta}^{(e)})(\theta^{(e)} - \hat{\theta}^{(e)})^\top.\end{aligned} \tag{5}$$

The overall posterior approximation is given by:

$$\theta_{\text{SWAG}} | \mathcal{D} \sim \mathcal{N}\left(\theta_{\text{SWA}}, \frac{1}{2}(\Sigma_{\text{diag}} + \Sigma_{\text{low-rank}})\right). \tag{6}$$

Once the posterior distributions are approximated, the model is used at test time by sampling from these approximations. Specifically, we used $T = 80$, $L = 4$ and we dropped the learning rate by 25% every 20 epochs. After training, we drew $M = 4$ samples from the approximated posterior distributions and computed the average of the predicted distributions from these samples.

5.9. Improved variational online Gauss–Newton

Improved variational online Gauss–Newton (iVOGN) introduces an enhanced Bayesian learning algorithm tailored to address positive-definite constraints within the learning process [32]. This method is part of the variational inference domain where the posterior distribution is approximated by a simpler one $q(\theta|\omega)$, where ω are the parameters. However, the parameters ω most often require to satisfy constraints. For example, when a multivariate Gaussian variable is used for such approximation, the covariance matrix must be positive-definite. In this study, we assume the approximate distribution $q(\theta|\omega)$ to be a multivariate Gaussian distribution, where ω represents the average and the covariance matrix for the multivariate variable θ. The iVOGN method ensures the covariance matrix stays positive-definite throughout training. Additionally, all model parameters are incorporated in this approximation, capturing the model's complexity while adhering to the positive-definite constraint on the covariance matrix.

5.10. Stein variational gradient descent

Stein variational gradient descent (SVGD) is a gradient-based sampling algorithm for approximate inference [33]. Briefly, let the posterior distribution be $p(\theta|\mathcal{D})$. SVGD finds a set of n particles $\{z_i\}_{i=1}^{n}$ to approximate the posterior p. Each particle z is a vector containing the model parameters. The particles' "positions" are updated by the following expression:

$$z_i \leftarrow z_i + \epsilon \frac{1}{n} \sum_{j=1}^{n} \left[k(z_j, z_i) \nabla_{z_j} \log p(z_j|\mathcal{D}) + \nabla_{z_j} k(z_j, z_i) \right], \tag{7}$$

for all $i = 1, \ldots, n$, where ϵ is the step-size, and $k(z, z')$ is any positive definite kernel specified by the users, such as the radial basis function kernel $k(z, z') = \exp\left(-\frac{1}{h}\|z - z'\|_2^2\right)$, which can be thought of as encoding some similarity measure between different particles z. In this update, the term that contains the gradient of $\log p(z|\mathcal{D})$ drives the particles towards the high probability regions of $p(\theta|\mathcal{D})$, while the term with $\nabla_{z_j} k(z_j, z_i)$ acts as a repulsive force to push z_i away from z_j to avoid the particles from collapsing together. All model parameters were sampled using this technique, setting the SVGD's hyper-parameters to $n = 10$, $h = 10$ and $\epsilon = 0.001$.

5.11. Last layer Laplace approximation

A Laplace approximation (LA) is derived through a second-order Taylor expansion centered around the mode of a distribution [34]. The mode can be determined using conventional gradient-based methods or, as in our case, substituted with a local optimum found with gradient descent. Specifically, this is achieved by approximating the log posterior over the weights of a NN given a dataset $\mathcal{D}$ around the Maximum A Posteriori (MAP) estimate θ_{MAP}. Mathematically, this can be represented by the following expression.

$$\log p(\theta|\mathcal{D}) \approx \log p(\theta_{\mathrm{MAP}}|\mathcal{D}) - \frac{1}{2}(\theta - \theta_{\mathrm{MAP}})^{\top} \bar{H} (\theta - \theta_{\mathrm{MAP}}), \tag{8}$$

where θ are the model parameters, and $\bar{H}$ denotes the Hessian of the negative log posterior. The absence of the first-order term is due to the expansion around a maximum (θ_{MAP}), where the gradient is zero. Upon exponentiating this equation, it becomes evident that the right-hand side adopts a Gaussian functional form for θ, leading to the approximation of a normal distribution through integration. The posterior over the weights is approximated as:

$$\theta|\mathcal{D} \sim \mathcal{N}(\theta_{MAP}, \bar{H}^{-1}). \tag{9}$$

In our study, we applied the LA only to the final layer of our DL model; consequently, we referred to this variant as the "Last-Layer Laplace Approximation" (LLLA). We implemented LLLA using the `laplace-torch` library [35].

5.12. Training details

We used an NVIDIA A100 80GB GPU to run all methods using PyTorch 2.6.0. For training the DL model, we employed the Adam optimizer with a learning rate of 0.001 and a batch size of 128. The training process was limited to a maximum of 100 epochs, with early stopping used to prevent overfitting. Specifically, training was halted if the validation loss failed to improve for 6 consecutive epochs. By preventing further training beyond this stage, early stopping is known to mitigate the risk of overfitting to the training set by avoiding additional updates that could harm generalization. Given the imbalance in the dataset, we addressed this issue by incorporating focal loss [36] during training, with parameters set to $\alpha = 0.1$ and $\gamma = 2$.

6. Results

6.1. Evaluation metrics

We evaluated the performance of the models using key metrics, including sensitivity, specificity, F1-score, ECE, AUC-ROC (area under the curve-receiver operating characteristics) and negative log-likelihood (NLL).

Sensitivity and specificity are commonly used for assessing the performance of AF detection from the DL model. Sensitivity measures the ability of the model to correctly identify positive cases (AF), while specificity gauges the model's accuracy in identifying negative cases (Non-AF).

ECE is a measure of how well the predicted probabilities align with the actual outcomes. It measures the difference between the average predicted probability and the actual observed frequency of events across various confidence intervals. Lower ECE values indicate better calibration. Formally, ECE is computed as follows:

$$\begin{aligned} \mathrm{ECE} &= \sum_{z=1}^{Z} \frac{|B_z|}{N} \left|\mathrm{acc}(B_z) - \mathrm{conf}(B_z)\right| \\ \mathrm{acc}(B_z) &= \frac{1}{|B_z|} \sum_{i \in B_z} \mathbb{I}(\hat{y}_i = y_i) \\ \mathrm{conf}(B_z) &= \frac{1}{|B_z|} \sum_{i \in B_z} \hat{p}_i \end{aligned} \tag{10}$$

where $\mathbb{I}(\hat{y}_i = y_i)$ denotes the indicator function, which equals 1 if the predicted label $\hat{y}_i$ matches the true label y_i for the ith sample, and 0 otherwise. B_z is the set of samples whose confidence predicted by the model (*i.e.*, model's output probability) is in the interval $[z-1, z)/Z$ where Z represents the total number of bins. N is the total number of instances across all bins, $\mathrm{acc}(B_z)$ denotes the accuracy of the zth bin, and $\mathrm{conf}(B_z)$ refers to the average confidence score of the samples in the zth bin. $i \in B_z$ indicates a subset of instances that have similar confidence scores and are grouped together in the same bin. We set the total number of bins $Z = 10$.

NLL is a measure of how well model's predicted probabilities match the true distribution of the data. Lower NLL values suggest better alignment. Mathematically, NLL is formulated as:

$$\mathrm{NLL} = -\frac{1}{N} \sum_{i=1}^{N} \log(p_{i,y_i}), \tag{11}$$

where p_{i,y_i} denotes the predicted probability for instance i and the correct class y_i.

6.2. Comparative performance for different UQ methods

Table 2 presents a comparative performance of UQ methods applied to the test set of the LTAF dataset. Methods such as PE, BE, and SWAG stand out for their robust performance, characterized by high sensitivity, specificity, AUC-ROC, and well-calibration (low ECE). Additionally, these methods demonstrate competitive performance in terms of NLL,

Table 2
Performance for UQ methods on the test set of the LTAF dataset ("internal testing").

Model	Sensitivity	Specificity	F1-Score	AUC-ROC	ECE	NLL
Baseline	0.930	0.941	0.948	0.989	0.230	0.874
MCD	0.826	0.978	0.899	0.960	0.095	0.581
DE	0.834	0.978	0.903	0.960	0.092	0.575
SE	0.915	0.978	0.950	0.970	0.061	0.549
BE	0.951	0.988	0.972	0.991	0.020	0.481
PE	**0.996**	**0.994**	**0.996**	**0.992**	**0.007**	**0.438**
SWAG	0.941	0.948	0.955	0.960	0.051	0.543
MFVI	0.851	0.931	0.901	0.971	0.087	0.573
MFVI (rank-1)	0.885	0.941	0.923	0.971	0.077	0.513
SVGD	0.802	0.931	0.871	0.921	0.131	0.681
iVOGN	0.791	0.930	0.864	0.920	0.112	0.612
LLLA	0.923	0.946	0.945	0.992	0.089	0.531

The Baseline model refers to the deterministic DL classifier without any UQ mechanism.

Table 3
Performance for UQ methods on the test set of the IRIDIA-AF dataset ("internal testing").

Model	Sensitivity	Specificity	F1-Score	AUC-ROC	ECE	NLL
Baseline	0.936	0.932	0.837	0.921	0.091	0.949
MCD	0.934	0.938	0.846	0.962	0.055	0.747
DE	0.934	0.936	0.843	0.961	0.054	0.749
SE	0.935	0.927	0.829	0.962	0.054	0.712
BE	0.935	**0.949**	**0.866**	0.974	0.052	0.593
PE	0.937	0.942	0.855	**0.975**	**0.046**	**0.384**
SWAG	0.929	0.945	0.856	0.971	0.057	0.701
MFVI	0.833	0.841	0.658	0.896	0.082	0.811
MFVI (rank-1)	0.830	0.840	0.655	0.901	0.091	0.823
SVGD	0.840	0.850	0.672	0.891	0.083	0.645
iVOGN	**0.942**	0.881	0.763	0.949	0.112	1.072
LLLA	0.938	0.931	0.837	0.931	0.069	0.762

indicating their ability to provide accurate probabilistic predictions. In contrast, SVGD and MFVI exhibit comparatively weaker performance across these metrics, indicating more uncertainty, poorer calibration, and less accurate probabilistic predictions.

Additionally, Table 3 presents the performance of each UQ method on the IRIDIA-AF dataset. Similar to the results on the LTAF dataset, PE and BE exhibit high AUC-ROC scores and low ECE and NLL values, underscoring their superior discriminative ability and calibration. In contrast, models like SVGD and iVOGN show lower performance and higher uncertainty.

Fig. 2 presents reliability diagrams with 10 equal-width bins for all 11 UQ methods on the LTAF test set. These plots visually complement the ECE values by illustrating where and how calibration errors occur.

The baseline model (panel a) exhibits pronounced overconfidence, with an ECE of 0.230 and large deviations between confidence and accuracy in higher bins. Ensemble-based methods, such as MCD, DE, and SE (panels b–d) provide notable improvements, reducing calibration gaps across most confidence ranges. BE and PE (panels e–f) are especially effective, achieving ECEs below 0.020 and near-perfect alignment with the diagonal.

Variational approaches (panels h–i) provide more heterogeneous behavior. MFVI and its rank-1 variant achieve reasonable calibration ($ECE \leq 0.087$), while SWAG performs better, with an ECE of 0.051 and more consistent calibration across probability bins.

Finally, panels j–l yield mixed results. SVGD and iVOGN exhibit comparatively higher ECE values (0.131 and 0.112, respectively), whereas LLLA demonstrates a little bit better calibration.

Overall, the reliability diagrams enrich the scalar ECE metric by providing a visual diagnosis of calibration quality, highlighting whether methods are systematically over- or under-confident across different confidence levels.

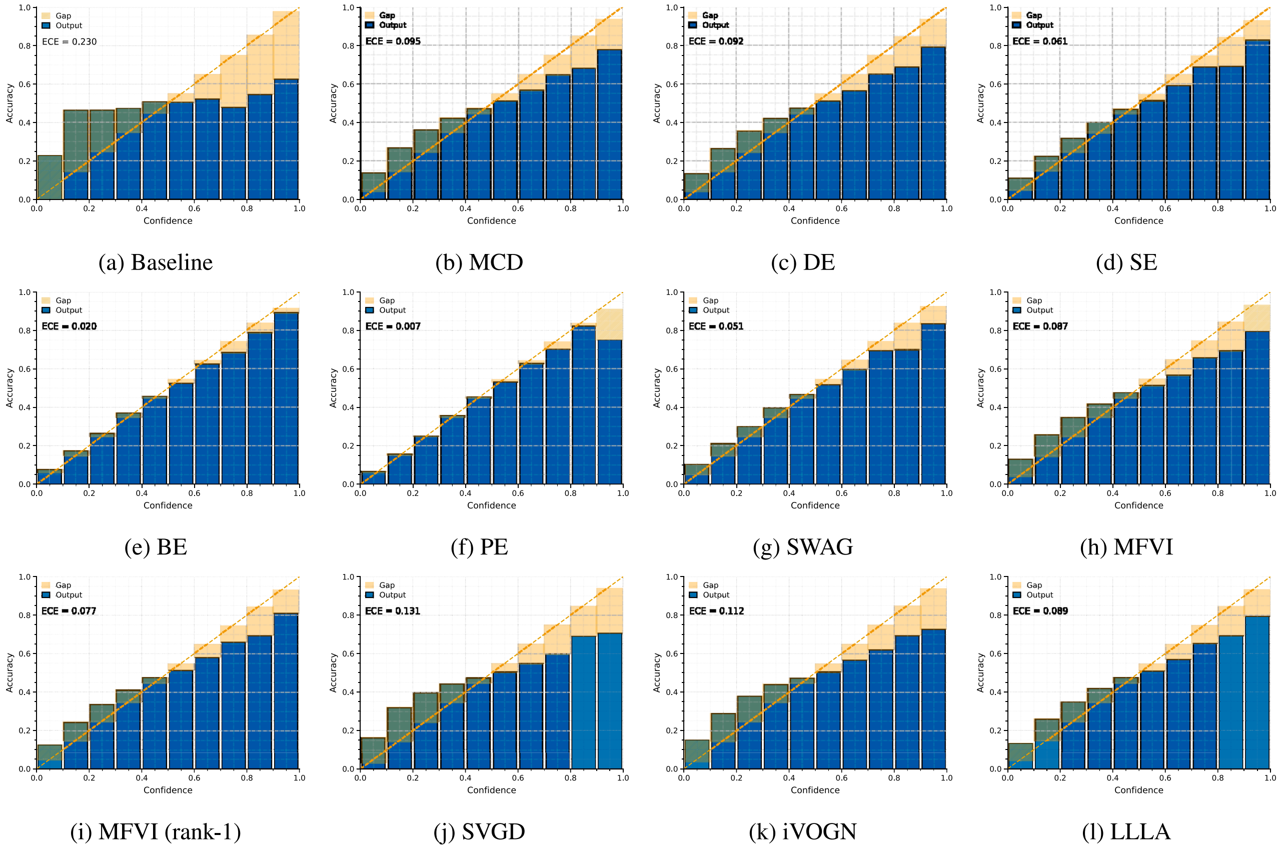
6.3. External validation

Our study underscores the critical importance of selecting UQ methods that maintain consistent performance across external test sets, particularly those trained on one dataset and tested on another. Table 4 presents the performance on various UQ methods trained on the LTAF dataset and tested on the MIT-BIH-AF (external) dataset (the test set comes from a different dataset than the train and validation sets). To ensure compatibility, the MIT-BIH-AF signals (originally sampled at 250 Hz) were resampled to 128 Hz, matching the sampling rate of the LTAF-trained models.

Considering all evaluated methods, the results reveal distinct tradeoffs between sensitivity, specificity, F1-score, calibration, and probabilistic prediction quality. Among the ensemble-based approaches, PE achieves the highest sensitivity (0.953) and the best AUC-ROC (0.944), underscoring its strong ability to detect AF cases. However, this comes with relatively low specificity (0.581), resulting in a moderate F1-score (0.726). iVOGN also achieves very high sensitivity (0.920), but its specificity (0.621) is lower, limiting precision. In contrast, LLLA provides the highest specificity (0.756), together with balanced sensitivity (0.895) and the strongest F1-score (0.783), although its NLL (0.732) suggests weaker calibration.

SE stands out for its excellent calibration, achieving both the lowest ECE (0.081) and lowest NLL (0.566), while maintaining solid sensitivity (0.894), specificity (0.648), and F1-score (0.727). Similarly, BE, SWAG, and SVGD show competitive performance, with relatively high specificities (0.632–0.703) and consistently low NLLs (0.578–0.601), supporting reliable probabilistic predictions. DE and MCD yield moderate results, with sensitivities around 0.89 but lower specificities (0.601–0.604), leading to weaker F1-scores (≈ 0.703–0.708) compared to SE, BE, or LLLA. Finally, MFVI and its rank-1 variant perform the weakest overall, with lower sensitivities (0.830–0.837), lower F1-scores (≈ 0.681–0.686), and comparatively higher calibration errors, indicating challenges in uncertainty modeling.

Overall, the specificity dropped from approximately 90% during internal validation to a range of 58.10% to 75.60% during external testing. These findings are consistent with the existing literature, such as the study by Seo et al. [37], which demonstrates that models trained on data from a specific source may not generalize well to external datasets, underscoring the adage "one-size-does-not-fit-all".



(a) Baseline (b) MCD (c) DE (d) SE

(e) BE (f) PE (g) SWAG (h) MFVI

(i) MFVI (rank-1) (j) SVGD (k) iVOGN (l) LLLA

Fig. 2. Reliability diagrams on the LTAF test set for different UQ methods. The horizontal axis corresponds to the predicted confidence, and the vertical axis indicates the accuracy. The orange diagonal represents perfect calibration, i.e., a one-to-one correspondence between confidence and accuracy. Blue bars denote the observed accuracy within each bin, while shaded regions illustrate the calibration gap. ECE values are provided in each panel as quantitative measures of miscalibration, with the corresponding numeric values reported in Table 2.

Table 4
Performance for UQ methods on the entire MIT-BIH-AF dataset (“external testing”).

Model	Sensitivity	Specificity	F1-Score	AUC-ROC	ECE	NLL
Baseline	0.893	0.752	0.780	0.911	0.159	0.752
MCD	0.887	0.602	0.703	0.862	0.149	0.722
DE	0.895	0.604	0.708	0.873	0.141	0.662
SE	0.894	0.648	0.727	0.851	0.081	**0.566**
BE	0.872	0.632	0.709	0.881	0.098	0.578
PE	**0.953**	0.581	0.726	**0.944**	**0.097**	0.582
SWAG	0.833	0.692	0.716	0.860	0.098	0.578
MFVI	0.830	0.632	0.686	0.859	0.117	0.612
MFVI (rank-1)	0.837	0.612	0.681	0.882	0.138	0.632
SVGD	0.825	0.703	0.717	0.871	0.010	0.601
iVOGN	0.920	0.621	0.728	0.854	0.113	0.642
LLLA	0.895	**0.756**	**0.783**	0.903	0.152	0.732

6.4. Impact of addition of random noise

By evaluating UQ methods under noisy conditions, our study is meant to verify their reliability in real-world environments where data may be corrupted. To achieve this, we added a white Gaussian noise to the ECG signal during inference, with standard deviations ranging from 0.01 to 0.055 mV with step of 0.005 mV. Within this range, the signal-to-noise ratio (SNR) varies from 9.6 to 23.5 dB, allowing an assessment of the UQ methods’ performance under different noise intensities. We selected a broadband Gaussian noise to degrade the signal quality, without focusing on specific interference types such as powerline noise, muscular artifacts, or baseline drifts. In fact, it is essential to highlight that these latter noise sources are typically mitigated during standard ECG preprocessing steps. For example, powerline interference and baseline drift are usually filtered out through bandpass filtering in the range of 0.5 to 40 Hz, a common practice in ECG signal processing. By applying this filtering during preprocessing, we ensure that such noise is substantially compensated from the ECGs before model training and evaluation, rendering its presence negligible when evaluating the effect of UQ techniques on the model’s performance. Therefore, in this study, we focused on evaluating the performance of UQ methods after applying the standard ECG preprocessing on our data, before adding the Gaussian noise.

The F1-score analysis, illustrated in Fig. 3, represents the performance of various UQ methods across different noise levels on the test set of the LTAF dataset, with the DL model trained on the LTAF training set. Notably, the PE model outperforms the other methods in this scenario. Furthermore, in Fig. 3, the SE, BE, and PE models demonstrate consistent performance across varying noise levels, indicating that noise has a substantial impact on these methods.

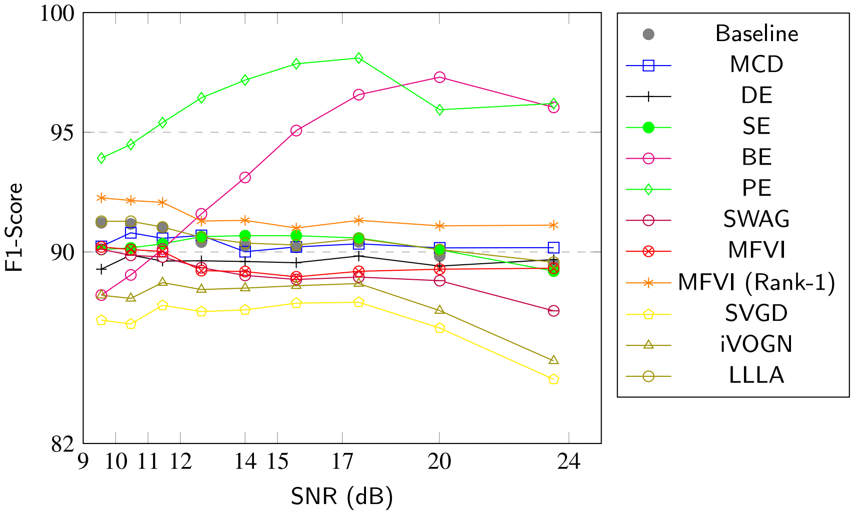


Fig. 3. F1-scores for different UQ methods when random noise is added to the signal.

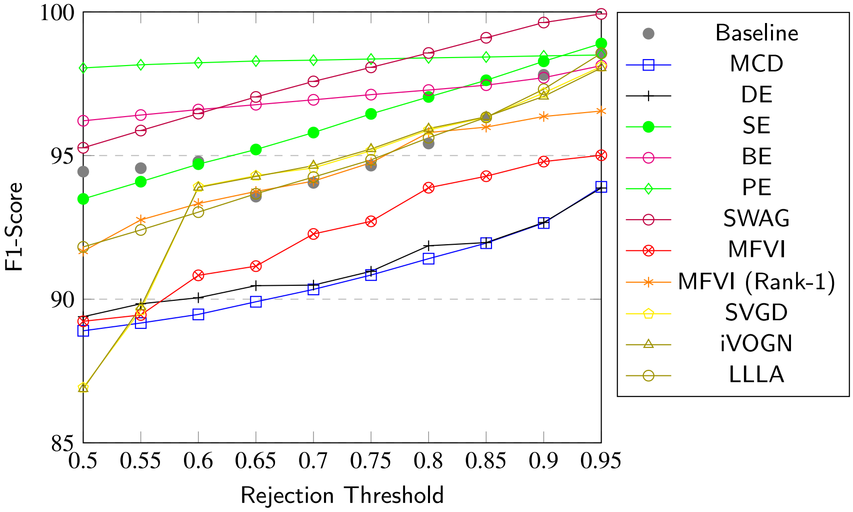


Fig. 4. F1-score for different UQ methods at different rejection thresholds.

6.5. *Classification with a rejection thresholds*

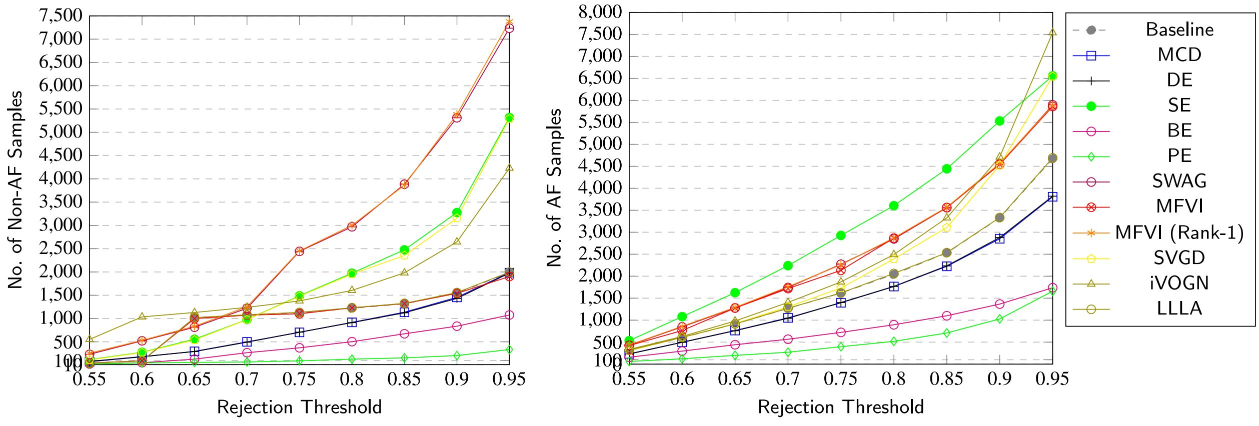
Classification with a rejection threshold, also known as reject inference, may enhance conventional classification models by enabling them to discard predictions when uncertainty is high. This approach is particularly beneficial in handling inputs that present classification challenges, as making low-confidence predictions could lead to errors.

In this study, we implement a decision threshold mechanism to assign class labels based on predicted probabilities. By varying the threshold from 0.55 to 0.95 with the interval of 0.05, we let the model to reject predictions when confidence is below the threshold. Our findings show that increasing the rejection threshold enhances the classifier's performance. Fig. 4 demonstrates the F1-score of various UQ methods across different rejection thresholds, revealing that higher thresholds improve overall performance with minimal impact on PE methods.

In Fig. 5, the rejection rate for each method pertains to the number of AF and Non-AF instances discarded at various thresholds. As the rejection threshold increases, the model becomes more confident in discarding instances, leading to higher rejection rates. Methods such as SE, SWAG, MFVI (rank-1), SVGD, and iVOGN exhibit high rejection rates for AF cases, indicating a strong sensitivity to uncertainty and a more aggressive approach. This conservatism reduces false positives but risks over-rejecting true AF cases, potentially missing some genuine AF instances. In contrast, methods like MCD, DE, and MFVI show a more balanced rejection rate, increasing steadily with higher thresholds. These methods provide a moderate trade-off, which rejects Non-AF instances while minimizing the risk of misclassifying true AF cases as Non-AF. On the other hand, BE, PE, and LLLA demonstrate a more conservative strategy, rejecting fewer AF cases compared to other methods. This cautious approach suggests a focus on reducing false negatives, ensuring that fewer true AF cases are incorrectly classified as Non-AF. Thus, selecting the appropriate method involves balancing the rejection of Non-AF instances with the risk of missing true AF cases, tailored to the specific needs and priorities of the application.

6.6. *Effect of ensemble size*

To assess the impact of ensemble size, we conducted additional experiments varying M across $2, 4, 8, 16, 32$. The results (see Fig. 6) show that performance consistently improves from $M = 2$ to $M = 4$, but additional gains quickly plateau, with changes of less than 0.003 across sensitivity, specificity, and F1-score up to $M = 32$. This finding aligns with prior studies [38–40]. For example, Lakshminarayanan et al. [38] demonstrated that most benefits of deep ensembles arise with small to moderate sizes; Ovadia et al. [39] reported that 5–10 models



(a)

(b)

Fig. 5. Number of samples discarded for different UQ methods under different rejection thresholds. (A) No. of Non-AF samples. (B) No. of AF samples. In the test set of the LTAF dataset, the total number of Non-AF and AF samples is 14,991 and 26,839, respectively.

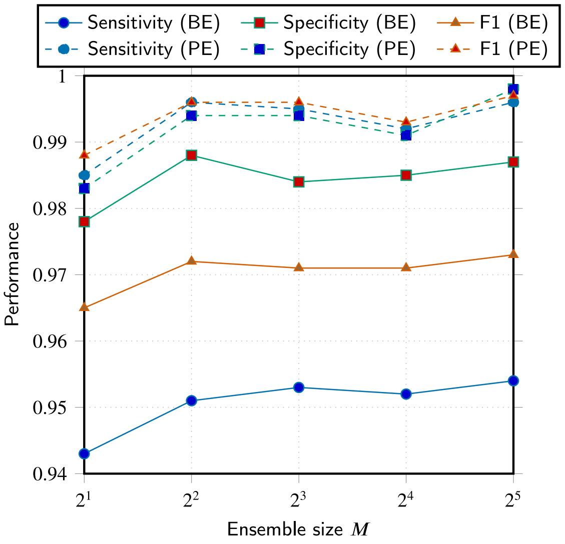


Fig. 6. Performance as a function of ensemble size. Sensitivity (circles), specificity (squares), and F_1-score (triangles) on the LTAF test set for the two top-performing methods (BE and PE) as the ensemble size increases ($M \in \{2, 4, 8, 16, 32\}$). Solid lines denote BE; dashed lines denote PE.

are typically sufficient for reliable predictive uncertainty; and Fort et al. [40] explained from a loss landscape perspective that relatively small ensembles capture the majority of diversity benefits. Given the linear computational cost of larger ensembles, we therefore select $M = 4$ as a fair and efficient choice for standardized benchmarking.

6.7. Efficiency of UQ methods

In addition to assessing performance metrics, our study explores the practical efficiency of various UQ methods by evaluating trainable parameters, inference time, and floating-point operations (FLOPs). The latter was estimated using the Python package fvcore [41]. Understanding the computational efficiency of these methods is crucial for their integration into clinical workflows, where resource constraints are prevalent.

Table 5 provides an evaluation of UQ techniques for efficiency on the LTAF dataset. Among these methods, MCD and DE demonstrate moderate inference times, each equipped with one million parameters and 5056 billion FLOPs. Notably, the SE stands out due to its expedited inference time, marginally reduced parameter count (0.99 million), and significantly lower FLOPs (1,686 billion), making it particularly suitable for efficiency-oriented applications.

MFVI (rank-1) exhibits an increase in both parameters and FLOPs. While SVGD, iVOGN, and LLLA show competitive inference times, their performance metrics discussed in the previous sections, including sensitivity, specificity, F1-score, AUC-ROC, ECE, and NLL, are less promising. Conversely, PE excels in efficiency, boasting an exceptionally fast inference time of 3.36×10^{-5} s, a modest 0.07 million parameters, and minimal 130 billion FLOPs.

Similarly, BE achieves an optimal balance, demonstrating a reasonable inference time of 9.48×10^{-5} s, 0.26 million parameters, and 421 billion FLOPs. This notable performance, particularly with respect to both speed and model complexity, highlights the advantages of PE and BE over other methods in our comparative analysis. Their efficiency makes them prominent candidates in scenarios where resource optimization is of paramount importance.

7. Discussions

This study conducts a comprehensive analysis of different UQ methods, focusing specifically on their application in AF detection. Acknowledging the pivotal role of uncertainty awareness in clinical decision-making, our objective is to identify the most suitable UQ methods for this purpose.

PE and BE consistently performed well across the IRIDIA-AF, LTAF, and MIT-BIH-AF datasets, demonstrating high sensitivity, specificity, and AUC-ROC values. These methods also provided low ECE and competitive NLL scores, reflecting their robustness and calibration abilities. This aligns with findings from previous studies which suggest that methods incorporating probabilistic approaches or ensembles, such as Bayesian methods and perturbation-based techniques, can offer enhanced reliability [10,42].

SVGD and MFVI showed comparatively weaker performance, especially in terms of sensitivity and calibration. These methods faced challenges in aligning predicted probabilities with true outcomes, resulting in higher ECE and NLL values. The less effective performance of SVGD and MFVI is consistent with the literature, indicating that methods relying on variational inference or optimization techniques might struggle with calibration and probabilistic accuracy [43,44].

The drop in performance when models trained on the LTAF dataset were evaluated on the MIT-BIH-AF dataset underscores a common issue in machine learning and medical diagnostics: model generalizability. The significant decrease in sensitivity from internal to external validation highlights the challenge of achieving robust performance across

Table 5
Average efficiency of UQ methods when applied on a 10-s ECG segment taken from the test set of the LTAF dataset.

Model	Inference time (s)	Parameters (10^6)	FLOPs (10^9)
MCD	1.33e−4	1.00	5056
DE	1.33e−4	1.00	5056
SE	8.00e−5	0.99	1686
BE	9.48e−5	0.26	421
PE	3.36e−5	0.07	130
SWAG	3.20e−5	0.25	421
MFVI	3.17e−5	0.25	1264
MFVI (rank-1)	1.89e−4	0.52	842
SVGD	3.23e−5	0.25	421
iVOGN	3.23e−5	0.25	421
LLLA	3.17e−5	0.25	1264

different datasets. This finding emphasizes the need for models to be trained on diverse datasets to improve their generalizability and reduce the risk of overfitting to specific data characteristics [45].

Our noise robustness analysis demonstrated that methods like PE maintained superior performance compared to others when subjected to varying levels of noise. This finding supports the use of PE in real-world applications where data corruption is a common concern. Conversely, methods like MCD and DE, while robust, exhibited higher computational demands, which may limit their practical applicability in scenarios with limited resources [9]. This is consistent with studies suggesting that while certain methods provide robustness, they may come at the cost of increased computational complexity.

Implementing rejection thresholds improved performance metrics, particularly sensitivity and specificity, as similarly obtained in other studies [46,47]. This approach allows for high-confidence predictions while avoiding uncertain cases, thereby potentially reducing error rates. However, it also presents a trade-off between rejecting too many instances and missing true positives. In this perspective, PE and BE showed to reject the least number of samples with respect to the other methods, supporting their use in clinical practice as well.

In terms of computational efficiency, PE and BE emerged as the most balanced methods, offering both high performance and low computational overhead. PE, in particular, stood out for its minimal inference time and parameter count, making it highly suitable for scenarios with stringent resource constraints. This is particularly relevant given the practical constraints of deploying UQ methods in clinical settings where real-time performance is crucial.

Overall, our comprehensive analysis of diverse UQ methods in AF detection highlights the multifaceted considerations essential for their effective application in clinical settings. By evaluating these methods through external validation, robustness under noise, classification with rejection options, and computational efficiency, we provide a holistic view of their strengths and limitations. Notably, while methods like PE and BE demonstrate superior efficiency and performance consistency, their integration into clinical workflows must be carefully balanced with the specific requirements of sensitivity, specificity, and computational resources.

8. Conclusion

In this study, we believe we made a significant contribution to the field of AF detection through a comprehensive investigation of UQ methods. Firstly, we examined 11 distinct UQ methods specifically tailored for AF detection using Holter recording data. Secondly, we conducted a rigorous evaluation of these UQ methods by introducing noise into the ECG data and evaluating its impact. This analysis not only evaluated the robustness of the UQ methods but also underscored their practical applicability in noisy environments. Finally, by analyzing the performance of the UQ techniques across various rejection thresholds, we provided valuable insights into their reliability and robustness. These detailed assessments aid in understanding the strengths and limitations of each method, facilitating more informed decision-making in clinical settings. Overall, our study advances the understanding of UQ methods in AF detection, paving the way for the development of more accurate and reliable diagnostic tools.

CRediT authorship contribution statement

Md Moklesur Rahman: Writing – original draft, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Massimo Walter Rivolta:** Writing – review & editing, Supervision, Investigation, Formal analysis. **Fabio Badilini:** Writing – review & editing, Validation, Supervision, Investigation. **Roberto Sassi:** Writing – review & editing, Validation, Supervision, Investigation.

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.

Acknowledgments

Part of this study was supported by the project FAIR (Future Artificial Intelligence Research) project, funded by the NextGenerationEU program within the PNRR-PE-AI scheme (M4C2, Investment 1.3, Line on Artificial Intelligence).

Data availability

Data will be made available on request.

References

[1] K.W. Davidson, M.J. Barry, C.M. Mangione, M. Cabana, A.B. Caughey, E.M. Davis, K.E. Donahue, C.A. Doubeni, J.W. Epling, M. Kubik, et al., Screening for atrial fibrillation: US preventive services task force recommendation statement, JAMA 327 (4) (2022) 360–367.

[2] A. Galli, F. Ambrosini, F. Lombardi, Holter monitoring and loop recorders: from research to clinical practice, Arrhythmia Electrophysiol. Rev. 5 (2) (2016) 136.

[3] G. Hindricks, T. Potpara, N. Dagres, E. Arbelo, J.J. Bax, C. Blomström-Lundqvist, G. Boriani, M. Castella, G.-A. Dan, P.E. Dilaveris, et al., 2020 ESC guidelines for the diagnosis and management of atrial fibrillation developed in collaboration with the European association for cardio-thoracic surgery (EACTS) the task force for the diagnosis and management of atrial fibrillation of the European society of cardiology (ESC) developed with the special contribution of the European heart rhythm association (EHRA) of the ESC, Eur. Heart J. 42 (5) (2021) 373–498.

[4] A. Rizwan, A. Zoha, I.B. Mabrouk, H.M. Sabbour, A.S. Al-Sumaiti, A. Alomainy, M.A. Imran, Q.H. Abbasi, A review on the state of the art in atrial fibrillation detection enabled by machine learning, IEEE Rev. Biomed. Eng. 14 (2020) 219–239.

[5] M.M. Rahman, M.W. Rivolta, M. Vaglio, P. Maison-Blanche, F. Badilini, R. Sassi, Residual-attention deep learning model for atrial fibrillation detection from Holter recordings, J. Electrocardiol. 89 (2025) 153876.

[6] M.M. Rahman, M.W. Rivolta, F. Badilini, R. Sassi, A systematic survey of data augmentation of ECG signals for AI applications, Sensors 23 (11) (2023) 5237.

[7] M. Gavidia, H. Zhu, A.N. Montanari, J. Fuentes, C. Cheng, S. Dubner, M. Chames, P. Maison-Blanche, M.M. Rahman, R. Sassi, F. Badilini, Y. Jiang, S. Zhang, H.-T. Zhang, H. Du, B. Teng, Y. Yuan, G. Wan, Z. Tang, X. He, X. Yang, J. Goncalves, Early warning of atrial fibrillation using deep learning, Patterns (2024) 100970.

[8] A. Kendall, Y. Gal, What uncertainties do we need in Bayesian deep learning for computer vision? Adv. Neural Inf. Process. Syst. 30 (2017).

[9] Y. Wen, D. Tran, J. Ba, BatchEnsemble: an alternative approach to efficient ensemble and lifelong learning, 2020, arXiv preprint arXiv:2002.06715.

[10] O. Laurent, A. Lafage, E. Tartaglione, G. Daniel, J.-M. Martinez, A. Bursuc, G. Franchi, Packed-Ensembles for efficient uncertainty estimation, 2022, ArXiv:2210.09184.

[11] M. Dusenberry, G. Jerfel, Y. Wen, Y. Ma, J. Snoek, K. Heller, B. Lakshminarayanan, D. Tran, Efficient and scalable Bayesian neural nets with rank-1 factors, in: International Conference on Machine Learning, 2020, pp. 2782–2792.

[12] W.J. Maddox, P. Izmailov, T. Garipov, D.P. Vetrov, A.G. Wilson, A simple baseline for bayesian uncertainty in deep learning, Adv. Neural Inf. Process. Syst. 32 (2019).

[13] J. Belen, S. Mousavi, A. Shamsoshoara, F. Afghah, An uncertainty estimation framework for risk assessment in deep learning-based AFib classification, in: 2020 54th Asilomar Conference on Signals, Systems, and Computers, 2020, pp. 960–964.

[14] A.O. Aseeri, Uncertainty-aware deep learning-based cardiac arrhythmias classification model of electrocardiogram signals, Computers 10 (6) (2021) 82.

[15] J.F. Vranken, R.R. van de Leur, D.K. Gupta, L.E. Juarez Orozco, R.J. Hassink, P. van der Harst, P.A. Doevendans, S. Gulshad, R. van Es, Uncertainty estimation for deep learning-based automated analysis of 12-lead electrocardiograms, Eur. Hear. Journal-Digital Health 2 (3) (2021) 401–415.

[16] B. Chen, G. Javadi, A. Hamilton, S. Sibley, P. Laird, P. Abolmaesumi, D. Maslove, P. Mousavi, Quantifying deep neural network uncertainty for atrial fibrillation detection with limited labels, Sci. Rep. 12 (1) (2022) 20140.

[17] Y. Elul, A.A. Rosenberg, A. Schuster, A.M. Bronstein, Y. Yaniv, Meeting the unmet needs of clinicians from AI systems showcased for cardiology with deep-learning–based ECG analysis, Proc. Natl. Acad. Sci. 118 (24) (2021) e2020620118.

[18] J. Park, K. Lee, N. Park, S.C. You, J. Ko, Self-attention LSTM-FCN model for arrhythmia classification and uncertainty assessment, Artif. Intell. Med. 142 (2023) 102570.

[19] V. Jahmunah, E. Ng, R.-S. Tan, S.L. Oh, U.R. Acharya, Uncertainty quantification in DenseNet model using myocardial infarction ECG signals, Comput. Methods Programs Biomed. 229 (2023) 107308.

[20] M. Moklesur Rahman, M. Rivolta, P. Maison Blanche, F. Badilini, R. Sassi, et al., Evidential deep learning model for atrial fibrillation detection from Holter recordings, in: Computing in Cardiology, 2024, pp. 1–4.

[21] M.M. Rahman, M.W. Rivolta, F. Badilini, R. Sassi, Quantifying uncertainty of a deep learning model for atrial fibrillation detection from ECG signals, in: 2023 Computing in Cardiology, vol. 50, 2023, pp. 1–4.

[22] G. Huang, Y. Li, G. Pleiss, Z. Liu, J.E. Hopcroft, K.Q. Weinberger, Snapshot ensembles: Train 1, get m for free, 2017, ArXiv:1704.00109.

[23] M. Abdar, F. Pourpanah, S. Hussain, D. Rezazadegan, L. Liu, M. Ghavamzadeh, P. Fieguth, X. Cao, A. Khosravi, U.R. Acharya, et al., A review of uncertainty quantification in deep learning: Techniques, applications and challenges, Inf. Fusion 76 (2021) 243–297.

[24] W. Zhang, X. Di, G. Wei, S. Geng, Z. Fu, S. Hong, A deep Bayesian neural network for cardiac arrhythmia classification with rejection from ECG recordings, 2022, ArXiv:2203.00512.

[25] S. Petrutiu, A.V. Sahakian, S. Swiryn, Abrupt changes in fibrillatory wave characteristics at the termination of paroxysmal atrial fibrillation in humans, Europace 9 (7) (2007) 466–470.

[26] C. Gilon, J.-M. Grégoire, M. Mathieu, S. Carlier, H. Bersini, IRIDIA-AF, a large paroxysmal atrial fibrillation long-term electrocardiogram monitoring database, Sci. Data 10 (1) (2023) 714.

[27] A.L. Goldberger, L.A. Amaral, L. Glass, J.M. Hausdorff, P.C. Ivanov, R.G. Mark, J.E. Mietus, G.B. Moody, C.-K. Peng, H.E. Stanley, PhysioBank, PhysioToolkit, and PhysioNet: components of a new research resource for complex physiologic signals, Circulation 101 (23) (2000) e215–e220.

[28] P. Baldi, P.J. Sadowski, Understanding dropout, Adv. Neural Inf. Process. Syst. 26 (2013).

[29] B. Lakshminarayanan, A. Pritzel, C. Blundell, Simple and scalable predictive uncertainty estimation using deep ensembles, Adv. Neural Inf. Process. Syst. 30 (2017).

[30] C. Zhang, J. Bütepage, H. Kjellström, S. Mandt, Advances in variational inference, IEEE Trans. Pattern Anal. Mach. Intell. 41 (8) (2018) 2008–2026.

[31] P. Izmailov, D. Podoprikhin, T. Garipov, D. Vetrov, A.G. Wilson, Averaging weights leads to wider optima and better generalization, 2018, ArXiv:1803.05407.

[32] W. Lin, M. Schmidt, M.E. Khan, Handling the positive-definite constraint in the Bayesian learning rule, in: International Conference on Machine Learning, 2020, pp. 6116–6126.

[33] Q. Liu, D. Wang, Stein variational gradient descent: A general purpose Bayesian inference algorithm, Adv. Neural Inf. Process. Syst. 29 (2016).

[34] H. Ritter, A. Botev, D. Barber, A scalable Laplace approximation for neural networks, in: International Conference on Learning Representations, ICLR 2018-Conference Track Proceedings, vol. 6, 2018, pp. 1–15.

[35] E. Daxberger, A. Kristiadi, A. Immer, R. Eschenhagen, M. Bauer, P. Hennig, Laplace redux–effortless Bayesian deep learning, in: NeurIPS, 2021.

[36] T. Lin, Focal loss for dense object detection, 2017, ArXiv:1708.02002.

[37] H.-C. Seo, S. Oh, H. Kim, S. Joo, ECG data dependency for atrial fibrillation detection based on residual networks, Sci. Rep. 11 (1) (2021) 18256.

[38] B. Lakshminarayanan, A. Pritzel, C. Blundell, Simple and scalable predictive uncertainty estimation using deep ensembles, in: Advances in Neural Information Processing Systems (NeurIPS), 2017.

[39] Y. Ovadia, E. Fertig, J. Ren, Z. Nado, D. Sculley, S. Nowozin, J.V. Dillon, B. Lakshminarayanan, J. Snoek, Can you trust your model's uncertainty? Evaluating predictive uncertainty under dataset shift, in: Advances in Neural Information Processing Systems (NeurIPS), 2019.

[40] S. Fort, H. Hu, B. Lakshminarayanan, Deep ensembles: A loss landscape perspective, in: International Conference on Learning Representations, ICLR, 2020.

[41] F. Research, fvcore, 2022, https://github.com/facebookresearch/fvcore/blob/main/docs/flop_count.md.

[42] X. Chen, Y. Li, Y. Yang, Batch-Ensemble stochastic neural networks for out-of-distribution detection, in: International Conference on Acoustics, Speech and Signal Processing, ICASSP, 2023, pp. 1–5.

[43] C. Guo, G. Pleiss, Y. Sun, K.Q. Weinberger, On calibration of modern neural networks, in: International Conference on Machine Learning, 2017, pp. 1321–1330.

[44] V. Kuleshov, N. Fenner, S. Ermon, Accurate uncertainties for deep learning using calibrated regression, in: International Conference on Machine Learning, 2018, pp. 2796–2804.

[45] L. Vasconcelos, B.P. Martinez, M. Kent, S. Ansari, H. Ghanbari, I. Nenadic, Multi-center atrial fibrillation electrocardiogram (ECG) classification using Fourier space convolutional neural networks (FD-CNN) and transfer learning, J. Electrocardiol. 81 (2023) 201–206.

[46] A. Filos, S. Farquhar, A.N. Gomez, T.G. Rudner, Z. Kenton, L. Smith, M. Alizadeh, A. De Kroon, Y. Gal, A systematic comparison of Bayesian deep learning robustness in diabetic retinopathy tasks, 2019, ArXiv:1912.10481.

[47] F.C. Ghesu, B. Georgescu, A. Mansoor, Y. Yoo, E. Gibson, R. Vishwanath, A. Balachandran, J.M. Balter, Y. Cao, R. Singh, et al., Quantifying and leveraging predictive uncertainty for medical image assessment, Med. Image Anal. 68 (2021) 101855.