Improving PPG Signal Classification with Machine Learning: The Power of a Second Opinion

Hamzeh Asgharnezhad
Internship in BML
University of New South Wales
Sydney, Australia
hamzeh.asgharnezhad@gmail.com

Afshar Shamsi
Concordia Ins. Inf. Sys. Eng.
Concordia University
Montreal, Canada
af_shams@encs.concordia.ca

Ivan Bakhshayeshi
BioMedical Machine Learning Lab.
University of New South Wales
Sydney, Australia
Bakhshayeshi@outlook.com

Roohallah Alizadehsani
Int. Sys. Research and Innovation (IISRI)
Deakin University
Geelong, Australia
r.alizadehsani@deakin.edu.au

Somayyeh Chamaani
Faculty of Electrical Engineering
K.N. Toosi University of Technology
Tehran, Iran
chamaani@kntu.ac.ir

Hamid Alinejad-Rokny
BioMedical Machine Learning Lab (BML)
University of New South Wales
Sydney, Australia
h.alinejad@unsw.edu.au

***Abstract*—Photoplethysmography (PPG) is a non-invasive technique that uses light to measure blood volume changes in tissues. It is widely used in clinical settings for monitoring vital signs such as heart rate, blood oxygen saturation, and blood pressure. To extract relevant information from PPG signals, such as peak detection and signal quality assessment, careful processing is required. Although traditional machine learning-based methods are capable of extracting useful information from PPG signals, they do not provide a measure of their confidence. In contrast, probabilistic machine learning approaches, such as Bayesian networks, can quantify uncertainty and provide estimates of prediction confidence. This would be beneficial for clinical decision-making and lead to more transparent and interpretable results. This paper proposes a new confidence-aware framework for Cuff-Less prediction of blood pressure using Monte Carlo Dropout (MCD) and Bayesian optimization techniques. Our results demonstrate that our approach outperforms simple MCD, providing more reliable predictions.**

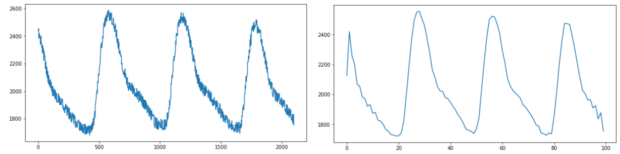
I. Introduction

Recent advancements in wearable technologies have made Photoplethysmography (PPG)-based screening/monitoring solutions accessible to a wider range of users ranging from athletes to patients in home care settings. Wearable devices such as smartwatches and fitness trackers are equipped with PPG sensors, allowing for continuous monitoring of heart rate, blood oxygen levels, and other physiological parameters. The increasing adoption of these devices has resulted in a growing need for accurate and reliable PPG signal processing algorithms that can handle large volumes of data in real-time [1]. Generally speaking, PPG signals are obtained via low-intensity infrared (IR) light targetted over specific body part and measuring the amount of light absorbed or reflected by the underlying blood vessels. PPG signals are a combination of several physiological processes, including cardiac activity, respiration, and autonomic nervous system activity. To diagnose medical conditions, various methods are developed to process PPG signals, including filtering, feature extraction, and Machine Learning (ML) algorithms [2]. In addition to medical applications, PPG signal processing has been increasingly used in sports and exercise science to monitor physical activity and recovery. Several studies have used PPG signals to measure heart rate variability and oxygen consumption during exercise, as well as to monitor recovery after physical activity. This information can be used to improve training programs and prevent overtraining, making PPG signal processing a valuable tool in sports science [3].

There has been a recent surge of interest in application of advanced ML algorithms, in particular Deep Neural Networks (DNNs)-based models, for processing PPG signals due to their ability to handle big data and extract relevant features, in an end-to-end fashion, to predict or diagnose medical conditions. For instance, DNN algorithms have been developed to predict blood pressure levels [4], and detect arrhythmia based on PPG signals [5]. DNN-based models have been shown to be effective in reducing the influence of noise and artifacts in PPG signals, and in adapting to changes in the signal over time [6], [7]. These features make DNN-based modeling a powerful tool for the analysis of PPG signals in clinical settings, particularly for large-scale studies involving thousands of patients or for remote monitoring applications. Although DNNs have shown promising results on difficult tasks, their reliability is vulnerable to unseen data, which can result in incorrect medical diagnoses and potentially fatal consequences for patients. This can be attributed to the fact that DNNs operate as black-box models with deterministic behavior, meaning that their decision-making process is opaque to humans, and they cannot accurately estimate their level of certainty regarding their predictions. The significance of quantifying uncertainty of DNN models has led to numerous studies in wide range of applications [8]–[11].

In this paper, we propose a novel technique for quantifying

979-8-3503-3959-8/23/$31.00 ©2023 IEEE



(a) PPG (b) Down sampled PPG

Fig. 1: This figure depicts two PPG signals. The left is the original one, and the right is the down sampled one to avoid fluctuations.

uncertainty in PPG signal processing using Monte Carlo Dropout (MCD) techniques. To the best of our knowledge, this is the first work to address uncertainty in PPG signals. Cuffless Blood Pressure (BP) estimation is, typically, performed by coupling PPG and ECG signals [12]. Recent works focus on BP estimation based on PPG signals alone, which is the focus of our framework. The ideal model should have the ability to communicate its level of confidence for various predictions, indicating whether it has sufficient confidence in its prediction by assigning high uncertainty to uncertain predictions and low uncertainty to certain ones. The proposed algorithm captures uncertain predictions, which should be further analyzed with a second opinion to ensure accurate decisions. Additionally, we introduce a new framework called Optimized Monte Carlo Dropout (OMCD), which can be optimized during the training phase. Our approach improves the reliability of predictions, enabling the model to handle its uncertainty more effectively. This is crucial for tasks dealing with human lives.

The rest of the paper is organized as follows: Section II presents the utilized dataset together with the background information about the proposed algorithm. The proposed framework is presented in Section III. The obtained results and simulations are provided in detail in Section IV. Finally, we summarize the study in Section V .

II. Background

In this section, first, the dataset utilized to develop the proposed farmework is presented. Next, the required background for developmed of the proposed framework is outlined.

A. Dataset

In this study, the main objective is to utilize the PPG signals to predict BP. To achieve this goal, the PPG-BP dataset [13] is used, which was recorded from 219 adults between the ages of 21 and 68. This dataset contains profiles of individuals and their health status in terms of blood pressure, and the research subjects are categorized into three groups, i.e., Normotension, Prehypertension, and Hypertension.The PPG signal was recorded while individuals were seated and relaxed, and appropriate methods were used to remove defective and high-noise signals, resulting in a dataset that only contains suitable PPG signals. The PPG signal for each individual was extracted using a sampling frequency of 1 kHz, resulting in $2,100$ signal points per person, equivalent to 2.1 seconds of data. An example of these signals is shown in the Fig. 1a. As per the recommendation of the data collection team, this study utilized the command “scipy.signal.resample” in Python, which utilizes the Fourier-based method, to down-sample the PPG signal. This was done to enhance the network’s performance and eliminate any extra fluctuations. Consequently, the PPG signals were reduced to 100 points per individual. Fig. 1b displays a minimized sample of the aforementioned signal.

B. Bayesian Machine Learning Uncertainty-aware classification involves considering two types of accuracies, one is related to model’s prediction ability and the other is related to the model’s uncertainty quantification capabilities. The rationale behind uncertainty quantification is that deterministic models are constrained in their ability to predict the class for an input, particularly if similar samples were not present during training. Depending solely on predictions can result in inaccurate outcomes in such cases. In contrast, uncertainty-aware models can assess their confidence in their predictions, which allows users to determine when to rely on the model’s predictions. The ideal model should have high accuracy in both uncertainty estimation and classification. Such a model would assign low uncertainty to correct predictions and high uncertainty to incorrect ones [14].

For uncertainty quantification, the Bayes formula is often used to perform Bayesian inference and estimate the posterior distribution of the model parameters given the observed data. The formula can be written as follows

$$p(Y|\mathcal{X},\theta) = \frac{p(\mathcal{X},Y|\theta)}{p(X|\theta)}, \tag{1}$$

where θ represents the model parameters, $\mathcal{X}$ represents the observed data, and Y denotes to labels. Term $p(\mathcal{X}|\theta)$ is known as marginal likelihood as well. In practice, it is often difficult to compute the denominator in Eq. (1) directly, and one usually resorts to approximations, such as Markov chain Monte Carlo (MCMC) or variational inference.

C. Variational Inference and Monte Carlo Dropout (MCD)

Variational inference is a technique used to approximate the posterior distribution of a probabilistic model, which is often intractable to compute exactly. The basic idea is to define a simpler parametric distribution, called the variational distribution, that closely approximates the true posterior. The parameters of the variational distribution are then optimized using an optimization algorithm such as gradient descent to minimize the distance between the variational distribution and the true posterior. This results in an approximate posterior that can be used for various purposes such as Bayesian inference or model selection.

Gal [15] proposed that dropout, a common regularization technique during training that randomly drops out certain neurons with a set probability, can be used as a tool for variational inference. This is achieved by enabling dropout during test time and forwarding multiple forward passes as

$$\mu = \frac{1}{T}\sum_{i=1}^{T} f(x_i, W), \tag{2}$$

where y is the output prediction, x is the input data, $f(\cdot)$ is the neural network function, W are the model parameters, and T is the number of Monte Carlo samples.

The intuition behind MCD is that by dropping out different neurons at each prediction, we are essentially exploring the different possible paths through the network. By averaging the predictions over all these paths, we get a more robust estimate of the model's output and an estimate of the uncertainty associated with that output.

Algorithm 1 OMCD

for M epochs **do**
  Train the model
  for T iterations **do**
    Perform forward pass with dropout:
    $\hat{y}_i = model(x_i)$
  end for
  Estimate PE Entropy for all $\hat{y}$
  Compute Loss = Cross Entropy + Mean of PEs
  Update weights by gradient descent
  Tuning the model's hyperparameter with BO
end for

III. Proposed Method

Motivated by [16] where MCD+entropy was proposed, in this section we develop the OMCD framework. To find the two identical distributions mentioned in Section II, we sort the output predictions of the underlying network by their Predictive Entropies (PEs), which could be estimated as follows

$$PE(x) = -\sum_{c=1}^{C} \mu_{(x,c)} \log\left[\mu_{(x,c)}\right] \tag{3}$$

where C is the number of classes and $\mu_{(x,c)}$ is the mean of predictions estimated by MCD algorithm, which is given by

The average PEs of all inputs in a batch will be added to the Cross Entropy loss function as follows

$$L = \sum_{b=1}^{B} \Bigg\{ \underbrace{-\left[y_b \log\left(\hat{\mu}_b\right) + (1-y_b)\log\left(1-\hat{\mu}_b\right)\right]}_{Cross\ Entropy} + \underbrace{\sum_{m=1}^{M} PE(x_b^{(m)})}_{Av.\ Entropy} \Bigg\} \tag{4}$$

where M is the number of forward passes of MCD, B is the batch size, C is the number of classes, y_b is the desired label for input sample x_b, and $\hat{Y}_{b,c}^{(m)}$ be a $[B \times C]$ matrix such that its b^{th} row is the network prediction (softmax output) corresponding to x_b in the m^{th} forward pass. In the MCD+entropy approach of Eq. (4), the objective is to estimate uncertainty of the model by incorporating PE into the loss function. The loss function defined in Eq. (4) has a significant advantage, as it allows the training procedure to backpropagate the entropies into the model. By minimizing the overall loss, the gradient descent algorithm strives to decrease the average entropies, which ultimately results in the model learning to reduce its entropic output. This process leads to more effective approah for capturing reliable uncertainity, i.e., being uncertainity-aware.

By leveraging the MCD concept, it becomes evident that the estimated predictions of the MCD model are heavily influenced by the chosen dropout probability, a key hyper-parameter that requires careful tuning prior to the training. Nevertheless, we present an optimized setup that utilizes Bayesian optimization techniques to efficiently determine the optimal dropout probability for various hidden layers of the model. The algorithm of OMCD is revealed in Algorithm 1. In this regard, the Uncertainty Accuracy (UA) metric proposed in [14] is selected as the objective function for the optimization step. UA measures the model's ability to distinguish between incorrect and correct predictions by assigning high uncertainty to wrong predictions and low uncertainty to correct ones. Furthermore, UA enables us to investigate performance of a Bayesian algorithm in one dimension despoite the fact that Bayesian networks typically involve both a measure of mean and a measure of variation. UA can be estimated as follows

$$UA = \frac{TC + TU}{\mathcal{X}}. \tag{5}$$

where TC denotes the number of inputs where the model has made an accurate prediction with certainty. On the other hand, TU indicates the number inputs in which the model has made an incorrect prediction and is extremely uncertain about them. Term $\mathcal{X}$ denotes to all input data. The above training algorithm helps the model to train with regard to its uncertainty and ultimately improve its performance for capturing uncertainty.

IV. Simulation and Results

In this paper, we extract the important features from the utilzied PPG signal dataset useng a one-dimensional convolution neural network. Generally speaking, one-dimensional convolution is a mathematical operation that involves sliding a small window, called a kernel/filter, over a one-dimensional input signal computing the dot product between the kernel and the values in the input signal that are currently overlapped by the kernel. The result of this operation is a new signal, often called the "convolved" signal, that reflects the degree of similarity between the input signal and the kernel at different locations. Reference [17] discusses the advantages of 1D convolution over traditional signal processing methods, and it provides examples of applications where 1D CNNs have shown superior performance. The features have been extracted using 2 convolution layers and fed to 2 hidden layers (using dropout regulizer) and a softmax on top of them to classify between three categories namely: Normtention (Normal), Prehypertension, and Hypertension.

Once the training is complete, the classification results for the test data are presented in the Table I below. As demonstrated by the results, the network is capable of classifying PPG signals with acceptable accuracy.

TABLE I: General performance of the model.

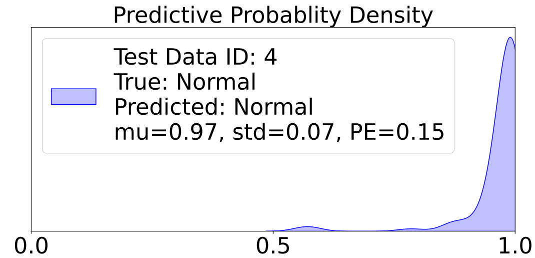
	Precision	Recall	F1-score	Support
Normal	0.85	0.83	0.84	60
Prehypertension	0.85	0.87	0.86	67
Hypertension	0.81	0.81	0.81	53
Overall	0.84	0.84	0.84	180

TABLE II: MCD's uncertainty confusion matrix which results in $UA = 83.4\%$.

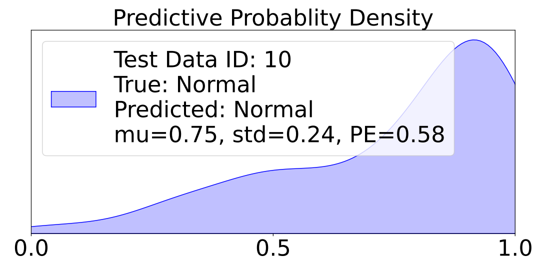
Uncertainty Confusion Matrix		Confidence	
		Certain	Uncertain
Correctness	Correct	TC=133	FU=18
	Incorrect	FC=12	TU=17

Having converted the model to its Bayesian variation [15] by keeping on the dropout at test time, the distributions of two sample inputs of Normtention (Normal class) are depicted in Fig. 2. Based on the style of the distribution in Fig. 2a, it is obvious that the model is certain about this prediction, since the distribution is located in the right part (larger than 0.5); however, the estimated distribution in Fig. 2b shows that although model classifies this input as Normtension (Normal), based on the distribution, the model would not assign it the same label for different runs (some points of the distribution are located at left corner, less than 0.5, which shows that the model is not certain). Here, we have reached a point where we cannot solely rely on the DNN model, and we require a second opinion. This may involve an in-person meeting with a clinician or obtaining additional information to accurately determine the patient's condition.

As stated previousely, in this paper we further improved the MCD algorithm by proposing its optimized variant called OMCD, which can be optimized during the training phase using Bayesian optimization. The performance of the MCD algorithm is highly dependent on the dropout rate, which is a hyperparameter that must be assigned by the developer. However, in the OMCD, this hyperparameter is automatically tuned during training, eliminating the need for manual assignment by the developer. This will help the model to assesss the predictions and try to assign low uncertainty for correct predictions and high uncertainty for wrong ones as much as possible. Table II shows the uncertainty confusion matrix of MCD (similar to [14]) for threshold equal to 0.5. The counterpart uncertainty confusion matrix of OMCD can be found in Table III. It is important to note that the optimized dropout rates for the first and second hidden layers of OMCD are 0.217 and 0.132, respectively. The UA, depicted in Table III with a value of 90%, shows improvement as the diagonal elements have increased. This suggests that the model has learned to evaluate its predictions and enhance its trustworthiness (i.e., improve its confidence) by assigning high prediction errors (PE) to incorrect predictions and low PE to correct ones.



(a) Normtension with low PE



(b) Normtension with high PE

Fig. 2: Distributions of two Normal inputs: (a) An illustrative example showing that the model classifies the test sample as Normtension with confidence. (b) Similar to (a) with the difference that the model is not confident in its prediction.

TABLE III: Uncertainty confusion matrix of OMCD and its corresponds UA equal to 90%.

Uncertainty Confusion Matrix		Confidence	
		Certain	Uncertain
Correctness	Correct	TC=144	FU=8
	Incorrect	FC=10	TU=18

V. Conclusion

In this paper, we apply a DNN model to PPG signals and obtain the posterior distribution of predictions using Bayesian inference. We demonstrate the importance of knowing confidence intervals and propose the OMCD technique as an improved alternative to the current MCD algorithm. The ability to determine confidence intervals is crucial in making reliable decisions for critical biomedical engineering tasks where uncertainty is inherently present, as it allows for a better understanding of the reliability and robustness of the model's predictions. In future work, we plan to employ various heuristic optimization algorithms, such as Whale and Grey Wolf, to determine if they can enhance our proposed method by conducting thorough comparative analyses of the results obtained using these algorithms to evaluate their efficacy and identify the best-suited one for our proposed method.

References

[1] J. Lin, R. Fu, X. Zhong, P. Yu, G. Tan, W. Li, H. Zhang, Y. Li, L. Zhou, and C. Ning, "Wearable sensors and devices for real-time cardiovascular disease monitoring," *Cell reports physical science*, vol. 2, no. 8, 2021.

[2] J. Allen, "Photoplethysmography and its application in clinical physiological measurement," *Physiological measurement*, vol. 28, no. 3, p. R1, 2007.

[3] L. Bosquet, S. Merkari, D. Arvisais, and A. E. Aubert, "Is heart rate a convenient tool to monitor over-reaching? a systematic review of the literature," *British journal of sports medicine*, vol. 42, no. 9, pp. 709–714, 2008.

[4] P. Santhanam and R. S. Ahima, "Machine learning and blood pressure," *The Journal of Clinical Hypertension*, vol. 21, no. 11, pp. 1735–1737, 2019.

[5] M. Sraitih, Y. Jabrane, and A. Hajjam El Hassani, "An automated system for ecg arrhythmia detection using machine learning techniques," *Journal of Clinical Medicine*, vol. 10, no. 22, p. 5450, 2021.

[6] E. J. Argüello Prada and R. D. Serna Maldonado, "A novel and low-complexity peak detection algorithm for heart rate estimation from low-amplitude photoplethysmographic (ppg) signals," *Journal of Medical Engineering & Technology*, vol. 42, no. 8, pp. 569–577, 2018.

[7] Q. Xie, G. Wang, Z. Peng, and Y. Lian, "Machine learning methods for real-time blood pressure measurement based on photoplethysmography," in *2018 IEEE 23rd International Conference on Digital Signal Processing (DSP)*. IEEE, 2018, pp. 1–5.

[8] M. Habibpour, H. Gharoun, A. Tajally, A. Shamsi, H. Asgharnezhad, A. Khosravi, and S. Nahavandi, "An uncertainty-aware deep learning framework for defect detection in casting products. arxiv 2021," *arXiv preprint arXiv:2107.11643*.

[9] M. Habibpour, H. Gharoun, M. Mehdipour, A. Tajally, H. Asgharnezhad, A. Shamsi, A. Khosravi, M. Shafie-Khah, S. Nahavandi, and J. P. Catalao, "Uncertainty-aware credit card fraud detection using deep learning," *arXiv preprint arXiv:2107.13508*, 2021.

[10] D. Khaledyan, A. Tajally, A. Sarkhosh, A. Shamsi, H. Asgharnezhad, A. Khosravi, and S. Nahavandi, "Confidence aware neural networks for skin cancer detection," *arXiv preprint arXiv:2107.09118*, 2021.

[11] A. Shamsi, H. Asgharnezhad, S. S. Jokandan, A. Khosravi, P. M. Kebria, D. Nahavandi, S. Nahavandi, and D. Srinivasan, "An uncertainty-aware transfer learning-based framework for covid-19 diagnosis," *IEEE transactions on neural networks and learning systems*, vol. 32, no. 4, pp. 1408–1417, 2021.

[12] S. Zabihi, E. Rahimian, F. Marefat, A. Asif, P. Mohseni, and A. Mohammadi, "Bp-net: Cuff-less and non-invasive blood pressure estimation via a generic deep convolutional architecture," *Biomedical Signal Processing and Control*, vol. 78, p. 103850, 2022.

[13] Y. Liang, Z. Chen, G. Liu, and M. Elgendi, "A new, short-recorded photoplethysmogram dataset for blood pressure monitoring in china," *Scientific data*, vol. 5, no. 1, pp. 1–7, 2018.

[14] H. Asgharnezhad, A. Shamsi, R. Alizadehsani, A. Khosravi, S. Nahavandi, Z. A. Sani, D. Srinivasan, and S. M. S. Islam, "Objective evaluation of deep uncertainty predictions for covid-19 detection," *Scientific Reports*, vol. 12, no. 1, pp. 1–11, 2022.

[15] Y. Gal and Z. Ghahramani, "Dropout as a bayesian approximation: Representing model uncertainty in deep learning," in *international conference on machine learning*. PMLR, 2016, pp. 1050–1059.

[16] A. Shamsi, H. Asgharnezhad, M. Abdar, A. Tajally, A. Khosravi, S. Nahavandi, and H. Leung, "Improving mc-dropout uncertainty estimates with calibration error-based optimization," *arXiv preprint arXiv:2110.03260*, 2021.

[17] S. Kiranyaz, O. Avci, O. Abdeljaber, T. Ince, M. Gabbouj, and D. J. Inman, "1d convolutional neural networks and applications: A survey," *Mechanical systems and signal processing*, vol. 151, p. 107398, 2021.