Received: November 2, 2022. Revised: December 15, 2022.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Arrhythmia Foetus Heartbeat Detection Using Optimized Neural Network based on Phonocardiograph Ensemble Feature and Principal Component Analysis

Irmalia Suryani Faradisa[1] **Oddy Virgantara Putra**[2]
Tri Arief Sardjono[1] **Mauridhi Hery Purnomo**[1]*

[1]*Department of Electrical Engineering, Institut Teknologi Sepuluh Nopember, Indonesia*
[2]*Department of Informatics, Universitas Darussalam Gontor, Indonesia*
* Corresponding author's Email: hery@ee.its.ac.id

Abstract: High-risk maternal health condition is alarming, especially in developing countries. Intensive monitoring is mandatory to prevent such issue. However, the long-term invasive method to pregnant women harms both the baby and the mother. In this research, we proposed a cost-efficient non-invasive foetal heartbeat classification based on a phonocardiograph with feature assembly. Since the high number of features and computationally expensive, we cut the size to half by utilizing Principal Component Analysis. Furthermore, data balancing using SMOTE is incorporated to improve classification performance. We proposed a method based on a neural network and optimized it using Random Search optimization. Eventually, the proposed method gained the top position in all data balancing compared to other machine learning algorithms, with 91.7 % for both accuracy and Area Under Curve with a score at 91.6 %.

Keywords: Arrhythmia, Dimensionality reduction, Ensemble feature, Neural network, Phonocardiograph.

1. Introduction

One target of the Sustainable Development Goals (SDG) is improving maternal health and reducing any threatening risks of pregnancy. The risks arise regarding lack of nutrition, blood issues, and disease-carrier mother [1, 2]. In developing countries, pregnant women are faced with high-risk pregnancies [3-5] This is mainly caused by inadequate information about maternal health awareness of blood-related disease [6, 7] Furthermore, due to minimum awareness, the risk spiked dramatically [6, 8]. All of these issues lead to an increase in the foetal mortality rate.

High-risk pregnancies demand such intensive monitoring using one or sometimes more of these devices, for instance, cardiotocography (CTG), Doppler Echocardiography (FED), and Fatal Electrocardiography (FECG). However, long-term usage of those might harm the foetal [9, 10].

In order to reduce such risk, many solutions have been conducted, mainly using a non-invasive method based on phonocardiography (PCG) [10-12]. PCG is a pure passive method of recording foetal heartbeat sounds. It can provide some crucial details regarding the foetal health situation, including the ability to find anomalies such as murmurs and intrauterine growth [10].

PCG is considered cost-efficient for heartbeat recording. However, a manual investigation and monitoring of foetal heartbeat are time-consuming. For these reasons, artificial intelligence (AI) approaches surge to the surface for tackling such issues [13, 14] Many works have been introduced using machine learning [15-18]. Their works were based on a single modal feature like MFCC. We proposed a framework for PCG abnormal heartbeat identification to address the issue.

The remaining of this paper are structured as follows: Related Works in Chapter 2, Proposed Model in Chapter 3, Results and Discussion in

Chapter 4, Conclusion in Chapter 5, and in the last chapter is Acknowledgement.

Our contribution towards the research of arrhythmia detection is in the feature size reduction while improving the performance of classification results and the neural network architecture optimization. Our feature combines audio features such as Mel-Frequency Cepstral Coefficients (MFCC), Zero Crossing Rate (ZCR), Chroma, and Spectral. We took 13 MFCC coefficients and then calculated each coefficient's mean and standard deviation. In addition to MFCC, we incorporated ZCR, Chroma Short-Term Fourier Transform (STFT), Spectral Centroid, Spectral Bandwidth, and Spectral Roll-off. Thus, our feature size is 31. Due to the large size of the feature, we utilize dimensionality reduction using Principal Component Analysis (PCA) before being fed to the classifier algorithm. However, manually hand-picking the parameters of the classifier takes a toll. Thus, we employed hyperparameter tuning to find the best parameters.

2. Related works

In the last decade, many approaches have been proposed for PCG audio classification using machine and deep learning. In 2018, Yaseen [15] proposed an automatic heart abnormality detection based on PCG signal using SVM and Deep Neural Network. The features used here were MFCC and DWT. The dataset contained five classes, Normal, Aortic Stenosis, Mitral Stenosis, Mitral Regurgitation, Mitral Valve Prolapse.

Yadav [17] proposed a cardiac disease classification using machine learning based on PCG sound. All features were picked using *p*-values, which are based on discrimination. On signal pre-processing, the band was taken between 20 - 500 Hz. All frequencies above the maximum band were removed as they were considered noises. The feature extraction filters used here were zero crossing rate, energy entropy, roll-off, and spectral flux. Then, each filter product was analysed using mean and standard deviation. His work was compared with four different algorithms: Support Vector Machine with linear kernel, Random Forest, Naive Bayes with Gaussian distribution, and *k*-nearest neighbours. It was mentioned that the proposed prominent feature selection gained impressive results.

Baghel [20] proposed a CNN based on identifying cardiovascular diseases (CVD) from PCG. The balanced target contains several classes: Aortic Stenosis, Mitral Regurgitation, Mitral Stenosis, Mitral Valve Prolapse, and Normal. Before training, the dataset underwent an augmentation step containing pitch, speed, time shifting, and back sound deformity. By applying data augmentation, the number of the dataset was doubled. The proposed CNN architecture contains two one-dimensional convolutions with ReLU on each layer. The fully-connected (FC) layer has 128 nodes. The accuracy difference before and after data augmentation was increased by 2.4 %.

In the same year, Oh [13] proposed signal classification based on WaveNet. The dataset acquired from [15] contains five classes which similar to [20]. Each audio record was sampled with 8,000 Hz frequency and normalized from between -1 and 1. The WaveNet architecture consist of six residual blocks then followed by one-on-one convolution. For each residual block, there are two dilated one-dimension convolution. Finally, the FC layers contain ten and 5 nodes, respectively. The highest accuracy achieved here was 98.2 %.

In the next year, Mei [14] introduced a CVD audio classification based on wavelet scattering transform (WST). The dataset used is taken from Computing in Cardiology (CinC) [21] The proposed method was structured as Quality Assessment, Wavelet Scattering, Audio Classification, and Voting. The final step plays significant role for increasing the accuracy by two percent.

In Yadav work, it incorporated centroid and spectral features. Baghel's work, it directly used the signal fed to the CNN. Yaseen's works incorporated MFCC and DWT as features. Mei's work uses WST. In our work, we incorporated MFCC, Spetral, centroid features. Since the feature dimesion is large, we employ PCA.

3. Proposed work

In this part, we exhibit the overview of the proposed approach for the foetal heartbeat (FHB) sound classification. Our approach comprises multiple levels, such as audio analysis, ensemble feature extraction, data balancing, dimensionality reduction using PCA, and FHB abnormality detection. The audio analysis includes a visual plot of audio signals based on each feature in order to extract meaningful information such as audio pattern, peak, noise content, and waveform. In the feature extraction part, we employed two different features, MFCC and ensemble features constructed from MFCC, Zero Crossing Rate (ZCR), Chroma STFT, Spectral Centroid, Spectral Bandwidth, and Spectral Roll-off.

It is certain that in the medical world, particularly in pregnant women, abnormality

Received: November 2, 2022. Revised: December 15, 2022.

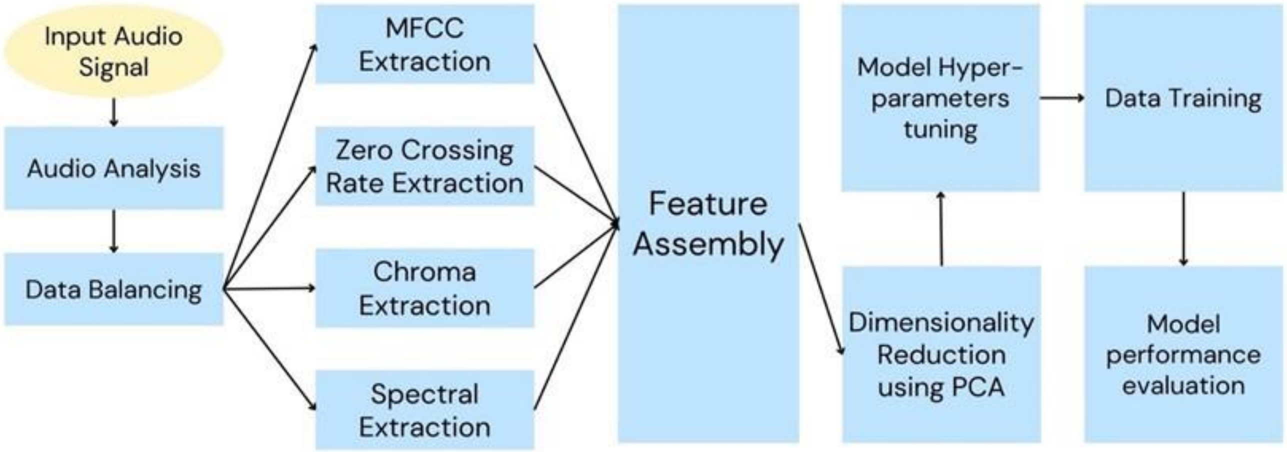


Figure. 1 The proposed framework block diagram

sometimes happens and is challenging to diagnose at the early stage of pregnancy. If an observation is taken, it leads to dataset imbalance. In order to tackle the such issue, we decided to employ several techniques, under-sampling, over-sampling, and synthetic minority oversampling technique (SMOTE). An imbalanced dataset is usually dominated by one class over the other. Here, we explored each method to determine the best for data balancing.

Under-sampling is one technique in data balancing. It works by reducing the amount of one or some majority classes until it meets the minority quantity. This method is incredibly fast since the number of the dataset is reduced. Since most data are removed, much information, probably the most important, has been lost. Meanwhile, for its counterpart, the number of minority data is arbitrarily duplicated to satisfy the majority class. Over-sampling is one of the favorable methods in data balancing. It can be helpful for machine learning methods where several duplicate samples can influence the model fit for a particular class if the distribution is skewed. However, due to its random behavior in selecting from minority data, over-sampling causes the probability of over-fitting increases significantly related to data variance. Such risk must be completely avoided.

To address the issue of both under-sampling and over-sampling, we exploit SMOTE. The concept of SMOTE is quite similar to under-sampling. It still utilizes data duplication out of the minor class. The difference lies in how data are duplicated. SMOTE nominates samples inside the feature set that are equivalent to one another, draws a line between the precedents, and then creates a new sample at a location along the line. In specific, stochastic specimens from a minor class are picked in the first place with *k* number of adjacent data. Subsequently, synthetic data is randomly generated between the two nearest neighbours. The position of the newly generated data is located inline.

In statistics, there are two well-known terminologies, univariate and multi-variate. These two resemble the number of features in the data. For one variable case, it is called single feature selection, and for multi variables, it is called multiple feature selection. For certain occurrences, features are probably taken from a single source. However, if the feature is collected from multiple sources, it is called a multi-modal. An audio signal can be extracted using distinct algorithms which decide their results. The outcomes can be discrete signals, spectrum, and chroma audios. For example, audio extracted using MFCC returns numerous features. They are called Mel features. Short-Time Fourier Transform (STFT) alters audio into a time-frequency domain using a frame-based signal of Fourier transform. STFT produces chroma features and spectrograms for signal visualization.

Principal Component Analysis PCA is a robust algorithm for handling issues in large-size features. Since our features are constructed from multiple features, -let us refer to this as an ensemble feature-, the feature size increases gradually. Due to numerous features, we decided to employ PCA to reduce the dimension by prioritizing only the most essential features.

Our data consist of 100 foetal heartbeat audio signals. A quarter of it contains abnormal heartbeats. According to this condition, our proposed method works like a charm from the aforementioned steps. The data are extracted using MFCC, STFT, and Zero Crossing. Before entering the feature reduction stage, they go into SMOTE to achieve balanced data. Subsequently, PCA took its part to prioritize important features only. All aforementioned steps are displayed in Fig. 1.

3.1 Audio pre-processing

This step plays an essential role mainly in achieving an outstanding classification performance. Thus, before being provided for further investigation, audio data must undergo a number of pre-processing steps. Data framing, the initial step, entails putting the audio data into a machine-readable format. After a certain amount of time, we gain value. The sampling rate is the pace at which it is sampled. For instance, we extract values every second from a 10-second audio recording. Here, we set the sampling rate (SR) to 44100 as default value.

Heartbeat sound is considered a non-stationary signal because its statistical attributes differ over a period. Accordingly, extracting the signal features such as spectral, short-time energy, and MFCC from the audio signal patch is imperative. The windowing process is crucial because it is based on the assumption that for every patch, the properties of the signal are stationary. The frame-blocking procedure is efficient for real-time systems over many samples.

Calculating the number of frames and samples is not rocket science. Assume there is an audio signal with a duration of five seconds and SR of 4 KHz. The SR is defined as the ratio between samples and signal per second. The SR value of 4 KHz must be converted into Hz. Now, we have 4,000 Hz per second. Thus, the samples are calculated as $4{,}000 * 5 = 20{,}000$ samples. On the contrary, the frame is a portion of the sample series.

The discrete signal is transformed into a time-frequency domain in the following stage. However, it is challenging for audio analysis, considering the time-frequency properties. Therefore, using STFT, we can visually inspect the audio signal using the spectrogram. A spectrogram is a visual representation of an audio signal with properties of time and amplitude. The x-axis of the spectrogram represents time, and the y-axis is the amplitude.

As we know, machine learning (ML) learns patterns from features. On a large scale feature, it is composed of a wide range of different data units. Each variable has its measurement and limit. If ML is fed with non-uniform range data, it is possible to suffer from dreadful recognition performance and lead to inconsistency. Therefore, we need the exact measurement for every single variable. To do this, we deployed a method called min-max scaling which can be attained by using Eq. (1):

$$\hat{x} = \frac{x - \min(x)}{\max(x) - \min(x)} \qquad (1)$$

where $\hat{x}$ is the input variable subtracted by the lowest possible number, then divided by the maximum and minimum possible number difference.

3.2 MFCC feature extraction

A raw audio signal holds much information. However, we can extract any necessary information by applying a particular technique. Here, we utilized the MFCC feature algorithm, which is widely used in speech and voice recognition. Generally, the steps in MFCC are pre-emphasis, frame blocking, frame windowing, fast Fourier transform, Mel spectrum, discrete cosine transform filter, and delta coefficient extraction.

Pre-emphasis contributes toward isolating high frequencies in order to balance the high slope roll-off spectrum of audio. Low-frequency signal has low variance in time, and the magnitude tends to move slowly. So that the part of the signal that has an insignificant change in adjacent windows is removed. Pre-emphasis is given by the following Eq. (2):

$$S(n) = s(n) - \alpha s(n-1) \qquad (2)$$

where $S(n)$ is the output signal, α is the controlling variable with range value between 0.9 and 1.0.

Frame blocking is also known as frame segmentation which generally splits the signal into 20~30 milliseconds frames. This process is done in a block that varies in terms of duration--the small value of the block assists in a detailed analysis. Nevertheless, the bigger window conveys a significant resolution of spectral signal. The results are multiplied using the Hamming window to preserve the constancy of the first and last points. The Hamming window is denoted as in Eq. (3):

$$w(n, \alpha) = -\alpha\left(1 + \cos\left(\frac{2\pi n}{N-1}\right)\right) \qquad (3)$$

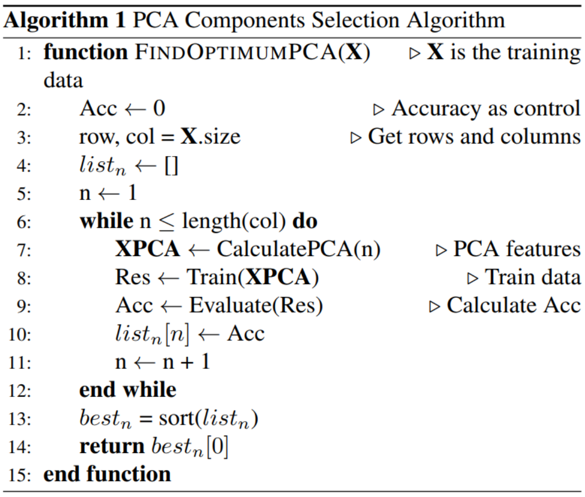
where n is the frame number, N is the total frame, and α affects the curve.

The products of each window from preceding calculation undergo to the Discrete Fourier Transform (DFT) to filter the magnitude. By using Eq. (4), we can get the result before entering the next process:

$$x(k) = \sum_{n=0}^{N-1} x(n) e^{\frac{-j2\pi kn}{N}} \qquad (4)$$

where $x(n)$ is the signal from frame blocking windows.

Algorithm 1. PCA components selection algorithm



```
Algorithm 1 PCA Components Selection Algorithm
function FINDOPTIMUMPCA(X)                ▷ X is the training data
    Acc ← 0                               ▷ Accuracy as control
    row, col = X.size                     ▷ Get rows and columns
    $list_n$ ← []
    n ← 1
    while n ≤ length(col) do
        XPCA ← CalculatePCA(n)            ▷ PCA features
        Res ← Train(XPCA)                 ▷ Train data
        Acc ← Evaluate(Res)               ▷ Calculate Acc
        $list_n[n]$ ← Acc
        n ← n + 1
    end while
    $best_n$ = sort($list_n$)
    return $best_n[0]$
end function
```

In the fifth step from MFCC feature extraction, a signal is distilled its spectrum using Mel-filter bank (MFB). A unit in Mel is measured based on human ear perception which has a better resolution on a lower frequency. Generally, MFB is applied for frequency and time domain. But, in this case, MFB only deployed in frequency domain. These formulae can be used to transform from Hertz (f) to Mel (m) using Eq. (5):

$$m = 2595 * \ln\left(1 + \frac{f}{700}\right) \tag{5}$$

and Mel to Hertz using Eq. (6):

$$f = 700(10^{\frac{m}{2595}} - 1) \tag{6}$$

As in Eq. (5), there is a natural log (ln). It is perceived that both log function and human ears have related properties. The input value (f) is dramatically increased but tends to be less at a higher value.

From Eq. (5), we applied the Mel frequency to Discrete Cosine Transform (DCT) to produce cepstral coefficients (CC). The system can be made robust by extracting only those coefficients, and truncating higher-order DCT components, as the first few MFCC coefficients represent the majority of the signal information. Therefore, we have:

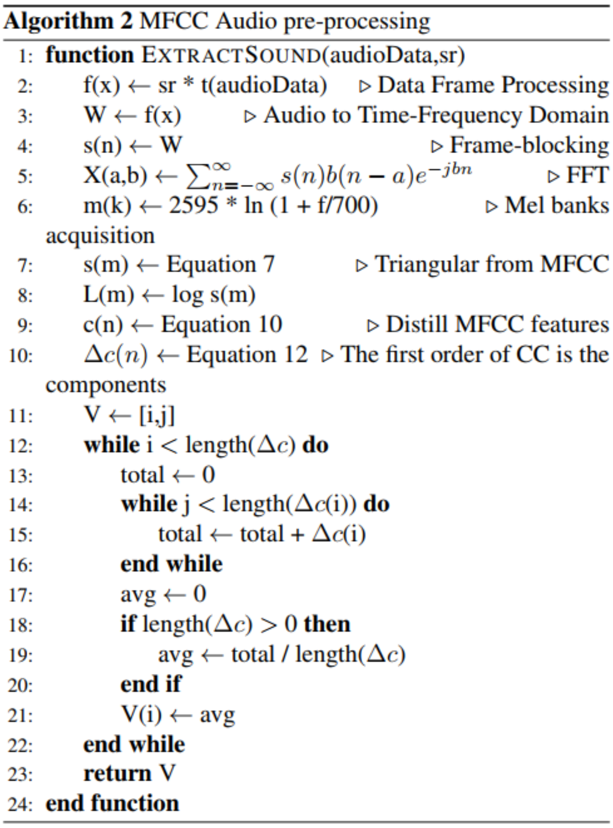
$$s(m) = \sum_{k=0}^{N-1}[|X(k)|^2 H_m(k)] \tag{7}$$

where the constraint of m is:

$$0 \leq m \leq M - 1 \tag{8}$$

then we calculate L by log base 10 from Eq. (7) using:

Algorithm 2. MFCC pre-processing



```
Algorithm 2 MFCC Audio pre-processing
function EXTRACTSOUND(audioData,sr)
    f(x) ← sr * t(audioData)              ▷ Data Frame Processing
    W ← f(x)                              ▷ Audio to Time-Frequency Domain
    s(n) ← W                              ▷ Frame-blocking
    X(a,b) ← $\sum_{n=-\infty}^{\infty} s(n)b(n-a)e^{-jbn}$    ▷ FFT
    m(k) ← 2595 * ln (1 + f/700)          ▷ Mel banks acquisition
    s(m) ← Equation 7                     ▷ Triangular from MFCC
    L(m) ← log s(m)
    c(n) ← Equation 10                    ▷ Distill MFCC features
    $\Delta c(n)$ ← Equation 12           ▷ The first order of CC is the components
    V ← [i,j]
    while i < length($\Delta c$) do
        total ← 0
        while j < length($\Delta c$(i)) do
            total ← total + $\Delta c$(i)
        end while
        avg ← 0
        if length($\Delta c$) > 0 then
            avg ← total / length($\Delta c$)
        end if
        V(i) ← avg
    end while
    return V
end function
```

$$L(m) = \log(s(m)) \tag{9}$$

Finally, we can measure Mel CC with:

$$c(n) = \sum_{m=1}^{M} \cos\left(\frac{\pi n(m-0.5)}{M+1}\right) L(m) \tag{10}$$

where the constraint n from Eq. (10) is:

$$n \in R | 0 \leq n \leq K - 1 \tag{11}$$

where $c(n)$ is the CC, and K is the number of CC. There are only 8-13 cepstral coefficients used in conventional MFCC systems. Since the zeroth coefficient indicates the mean of log energy from the original signal, which conveys a tiny amount of specific information, it is frequently removed. By this condition, we utilize 13 CCs.

The CC represents stationary attributes because of limited information from the input frame. To distill additional data, it is required to calculate the first and second derivative of CC. Thus, we can find the first-order using :

$$\Delta \mathrm{c_m}(n) = \frac{\left(\sum_{i=-T}^{T} k_i c_m(n+i)\right)}{\sum_{i=-T}^{T} |i|} \tag{12}$$

We can calculate the second order from Eq. (12).

Algorithm 3. Feature ensemble algorithm

```
Algorithm 3 Ensembled Feature and Data Balancing Algorithm
function ASSEMBLE($\Delta c$, C1, SC, SB, SR)
    M ← [$\Delta c$ C1 SC SB SR]                ▷ Feature Assembly
    baris, kolom = size(X)
    x ← []
    while i < length(kolom) do
        row ← X[:,i]
        x[i] ← count(row)
    end while
    $x_{min}$ ← min(x)
    T ← x.index($x_{min}$)                      ▷ smallest minority class
    N ← 100
    k ← 5                                       ▷ the number of nearest neighbor
    Smp ← []
    Syn ← []
    i ← 1
    while i ≤ T do
        Compute k for i, and put in nnarray
        Populate(N,i,nnarray)                   ▷ samples generators
    end while
    while N ≠ 0 do
        nn ← pick arbitrary number between 1 and k
        atr ← 1
        i ← 0
        idx ← 0
        while i < length(nnarray) do
            $\Delta d$ ← Smp[nnarray][nn][atr] - Smp[i][atr]
            gap ← rand(0,1)
            Syn[idx][atr] = Smp[i][atr] + gap * $\Delta d$
        end while
        idx ← idx + 1
        N ← N - 1
    end while
    X[x.index($x_{min}$)] ← Syn
    return X
end function
```

3.3 Ensemble feature selection and dimensional reduction

MFCC is decisive in human speech recognition due to its robust, high-resolution, low-frequency extraction. Since FHB has a low audio magnitude, relying only on MFCC is insufficient. Therefore, we exploit zero crossing rate, chroma, and spectral features to improve the audio classification. Each one of these additional features holds several unique properties. For example, we extract the centroid, bandwidth, and roll-off features in spectral. Then, they are assembled into the MFCC. We calculate the average of every single attribute and put it into a two-dimensional matrix **X** as follows:

$$X = \begin{bmatrix} cc_{11} & cc_{12} & \dots & cc_{1n} \\ cc_{21} & cc_{22} & \dots & cc_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ cc_{m1} & cc_{m2} & \dots & cc_{mn} \end{bmatrix} \quad (13)$$

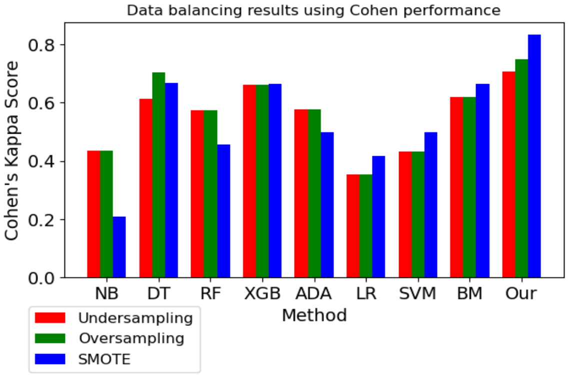


Figure. 2 Cohen performance results from our compared with other machine learning method

where m is the number of rows, n is the number of column features, and $n = 29$

The matrix **X** is quite large considering its property size and not all of those are essential. Handpicking and tuning to determine which feature affects the most to machine learning model is challenging and takes forever. There is no other way but to employ a dimensionality reduction method called Principal Component Analysis (PCA). PCA works by calculating the current features closeness towards each other into a new domain. Determining the dimension size in PCA is difficult. This component selection algorithm is presented in *Algorithm 1*.

3.4 Model for classification and parameter tuning setup

In this part, we proposed an optimized machine learning based on Neural Network. To prove the efficiency of our model, we compared with other traditional algorithms Random Forest (RF), Ada Boosting (ADA), Gradient Boosting (XGB), Log Regression (LR), Support Vector Machine (SVM), Decision Tree (DT), and Naive Bayes (NB).

*Algorithm 2*Algorithm exhibits the detailed step of MFCC extraction. Assume there is an audio signal T with the time duration t, where $f(x)$ is frame-blocking product of multiplication between sampling rate sr and t in line 2. Then, the frame $f(x)$ was altered to a time-frequency domain, and we get W. From W, the windowing result is represented by $s(n)$. Subsequently, $s(n)$ is transformed using FFT, and we acquired $\mathrm{X(a, b)}$, a two-dimensional matrix that holds both phase and magnitude. In line 6, we extracted the Mel-filter bank by taking the natural logarithmic of $1 + f/700$ multiplied by 2595. The triangular feature of MFCC $s(m)$ which is obtained using Eq. (7) then fed to Eq. (9) for further processing. Then, we took the log base 10 from $s(m)$ and we get $L(m)$. $L(m)$ is used as input in Eq. (10). Consequently, the cepstral coefficients are the delta of Δc, which is the first order from Eq. (12).

The Δc is a matrix with $m\ x\ n$ size where m is the length of CC length, and n is the number of CC. Instead of using all Δc, we calculate the mean from each CC coefficient. Then, we get a matrix **V** containing all the MFCC features we need. Before the final feature, we concatenate our MFCC features with others such as chroma and spectrum. Since our dataset is imbalance, we employ SMOTE as in *Algorithm 3*. Eventually, since each feature is numerical, we scaled our dataset using Eq. (1).

4. Experiments, results, and discussion

The study was carried out using a computer with Ubuntu 22.04 LTS, 16 GB RAM, processor intel core i7, GPU NVIDIA GTX 1050 Ti which contains 768 CUDA cores, Python version 3.7, and PyCharm Community Edition for the IDE.

The experiments are conducted as follows: dataset preparation, metric evaluator selection, Optimum PCA component candidate selection, baseline model, data balancing, and performance evaluation. All of these were delivered in order. Since our data is imbalanced and finding which one of the balancing methods suits well for our dataset, we utilized downsampling, oversampling, and SMOTE trials.

4.1 Dataset preparation

In this study, we evaluate a public fetal heartbeat audio dataset from Physionet using machine learning [9]. The dataset contains many audios recorded using Ultrasonography (USG) device. The sounds were focused on the lower abdomen of pregnant mothers in India. The results were 75 percent normal heartbeats, and the remaining were abnormal. All recordings have an average duration of 90 seconds, with the majority SR equal to 16 kHz, quantization at 16-bit, and the others at 44,100 Hz. A digital stethoscope was used to record with a wide-band setting and frequency between 20 to 1,000 Hz. We split the data into a 60:40 ratio for training and testing purposes.

4.2 Experiment setup and metrics evaluator

Here, we conduct experiments with seven ML algorithms in order to compare our model. Performance metrics used here are accuracy (ACC), precision (Prec), Recall (Rec), F1-score, Cohen's Kappa (CK), Receiving Operating Characteristic (ROC), and Area Under Curve (AUC). ACC is used because we need to evaluate the correctness between the predicted value of our model with the actual output value. Since it is not enough to measure the performance, we utilize Prec, defined as a ratio of the True Positive (TP) value with the sum of TP with the False Positive (FP) value. Prec is powerful when we need to find out how well our model corrected prediction and false prediction. Rec is especially useful when it is needed to categorize an event that has already happened. For instance, high Rec is required for fraud detection models to identify scams effectively. The genuine 0s are irrelevant to us in these circumstances because our primary goal is to identify the real 1s as frequently as feasible. The misidentification in Rec is called False Negative (FN). A good classifier is relied on how Prec and Rec are close to one. In other words, FP and FN tend to be zero. Therefore, the F1-score metric plays a role here.

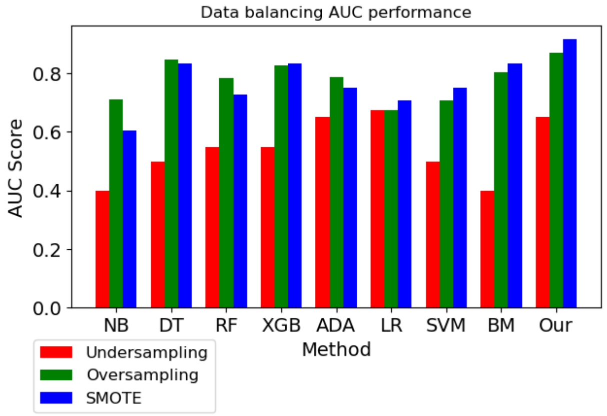


Figure. 3 AUC performance for each algorithm

4.3 PCA component selection

Due to the number of features in our dataset being quite large, the calculation is computationally expensive. To reduce the burden, we employ PCA. However, finding the optimum number of PCA components is expensive as well. Here, we utilize a brute force method in which the number of PCA components gained the best results as in Algorithm Fig. 2. First, let **X** be a matrix $m\ x\ n$ where the number of data and the current ensemble feature are represented by m and n, respectively. Second, we created a list to buffer the Accuracy, which is sorted in ascending order later. Third, we iterated all possible combinations of PCA based on the feature-length n. Before training, the PCA features (**XPCA**) are calculated, and then the Accuracy is measured. All Accuracy values are stored in a list. Eventually,

Table 1. Hyper-parameter with its optimum value

No.	Parameters	Value	Comment
1	Hidden Layer	3	layers
2	Nodes	50,100,50	Each layer
3	Activation Function	ReLU	-
4	Weight Optimizer	ADAM	-
5	Learning Rate	0.002	-

Table 2. Classification results from downsampling

No.	Model	Acc	Prec	Rec	F1-Score
1	NB	0.4	0.375	0.3	0.333
2	DT	0.5	0.5	0.7	0.583
3	RF	0.55	0.57	0.4	0.47
4	XGB	0.55	0.5333	0.8	0.64
5	ADA	0.65	0.636	0.7	0.66
6	LR	0.4	0.416	0.5	0.4545
7	SVM	0.5	0.5	0.3	0.374
8	BM	0.6	0.538	0.5	0.454
9	Our	0.65	0.636	0.7	0.667

Table 3. Classification results from oversampling

No.	Model	Acc	Prec	Rec	F1-Score
1	NB	0.729	0.846	0.5	0.628
2	DT	0.8125	0.933	0.636	0.756
3	RF	0.79	0.833	0.681	0.749
4	XGB	0.833	0.85	0.772	0.809
5	ADA	0.7916	0.8	0.727	0.761
6	LR	0.6875	0.733	0.5	0.594
7	SVM	0.729	**0.909**	0.454	0.606
8	BM	0.8125	0.842	0.727	0.78
9	Our	**0.8541**	0.8541	**0.8636**	**0.844**

we can get the best number of PCA components by sorting the list in ascending order and filtering by the zero indexes.

4.4 Multi-layered perceptron baseline model

Here, we tested and evaluated our proposed model with the baseline model (BM). The BM method is based on one of neural network descendant, the multi-layered perceptron (MLP). The BM hidden layer structure composed of two layers which each layers hold ten nodes. The learning rate was set to $1e-2$. Generally, when traning data using neural network, the loss error may fall into local minima. To tackle this issue, we set the activation function to ADAM.

4.5 The proposed model

One of the challenge in building a neural network (NN) model is finding the optimum hyper-parameter. By default, the structure is handpicked such as the number of layer, nodes, learning rate, and the activation function on each nodes. Of course, such work takes a toll on the time and energy. Hence, we conducted optimization (OPT) for hyper-parameters using Random Search. First, the number of hidden layer is considered in OPT because it is the foundation of an NN. Second, the activation function is also selected with varies among of Tanh, ReLU, Sigmoid, and Linear function. Third, the weight optimizer is picked from Quasi-Newtonian method, Stochastic Gradient Descent, and ADAM solver. The fourth parameter is the learning rate. We choose a variation from 0.001, 0.002, and 0.003. From Random Search, we got the optimized hyper-parameters as in Table 1.

4.6 Results from data balancing

The proposed work is compared with the BM and others, for example, NB, DT, RF, XGB, ADA, LR, and SVM. The comparison is evaluated in all three balancing methods. The bar chart in Fig. 3 displays the performance comparison of AUC for eight algorithms and the proposed method. The bar colors red, green, and blue represent the three distinguished data balancing methods: downsampling (DS), oversampling (OS), and SMOTE. Overall, we can see that the others overpower DS on average, and SMOTE is second to none. NB for DS is at the bottom tier with BM and is followed by SVM. Furthermore, LR and SVM are in the OS compared to DT, XGB, and RF in the lower section. At last, our model dominated all other methods for AUC scores in terms of DS and SMOTE.

Here, we also compared our method with other methods in Cohen's Kappa metrics. Cohen's Kappa is a measurement that can be used to assess the classification model. The assessment is based on the quantitative score of two raters and their agreement frequency. The graph in Fig. 2 exhibits the Cohen performance for every algorithm in three categories, DS, OS, and SMOTE. Our method stands in the top position in all categories, compared to others, followed by XGB, BM, and DT on average. Meanwhile, LR is in the lowest score in almost all categories.

After comparing the classification results using AUC and Cohen's Kappa, we also judge the prediction and actual results for our method with other algorithms. Table 2 shows the performance of all methods in Downsampling. At the same time, the performance results for Oversampling are in Table 3. Lastly, the SMOTE results are in Table 4.

From Table 2, our model achieved the top position for Acc, Rec, and F1-score with the scores

Table 4. Classification results from SMOTE

No.	Model	Acc	Prec	Rec	F1-Score
1	NB	0.604	0.5862	0.7083	0.6415
2	DT	0.833	0.86363	0.7916	0.826
3	RF	0.7291	0.866	0.5416	0.666
4	XGB	0.833	0.9	0.75	0.818
5	ADA	0.75	0.75	0.75	0.75
6	LR	0.708	0.6785	0.791	0.7307
7	SVM	0.75	0.8	0.66	0.727
8	BM	0.833	0.944	0.708	0.809
9	Our	**0.917**	**0.954**	**0.875**	**0.913**

Table 5. Performance comparison of SOTA and the proposed work

No.	Model	Feature	Acc (%)
1	YasSVM	MFCC+DWT	66.75
2	YasDNN	MFCC+DWT	95.33
3	YasKNN	MFCC+DWT	76.6
4	YadSVM	Centroid, Spectral, Entropy	73.5
5	YadNB	Centroid, Spectral, Entropy	67
6	YadRF	Centroid, Spectral, Entropy	95
7	YadKNN	Centroid, Spectral, Entropy	95.75
8	BaghelDNN	PCG Signal	93.75
9	MeiWST	WST	82
10	Our	Ensemble	**98.3**

at 85 %, 86 %, and 84%, respectively. Unfortunately, the Prec score is in the second position after DT. The wonderful performance of our model continued in Oversampling results. Our model scored 87.5 % in Acc, 86 % in Rec, and 86 % in F1-Score.

Finally, in the last balancing method score (SMOTE), our model overpowered all methods in all metrics. In terms of Acc, our model gained 91.7 % followed by BM, DT, and XGB, with a score of 83 % each. Meanwhile, we successfully secured the top position in Prec with a 1.1% difference between our and the second position, baseline model (BM). For Rec and F1-score, our model achieved 87.5% and 91.3 %, respectively.

4.7 Performance comparison with the state-of-the-art methods

In here, we compared our proposed work with the state-of-the-art (SOTA) algorithms Yaseen [15], Yadav [17], Baghel [20], and Mei [14]. Yaseen presented a public dataset which contains five classes of cardiac disease such as Aortic Stenosis (AS), Mitral Regurgitation (MR), Mitral Stenosis (MS), Mitral Valve Prolapse (MVP), and Normal (N). As our comparison, we employed four machine learning methods as in Yaseen's work and we coded each one of them as follows: SVM (YasSVM), Centroid KNN (YasKNN), and Deep Neural Network (YasDNN). As information, we only compared the best features (MFCC and DWT) based on the accuracy performance results in Yaseen work. In Yadav, we use his four machine learning works, SVM (YadSVM), Naïve Bayes (YadNB), Random Forest (YadRF), and k-Nearest Neighbour (YadKNN). The features used in Yadav are centroid, energy entropy, spectral roll-off, spectral flux, and zero crossing rate. Each result from Yadav features is calculated with mean and standard deviation to reduce complexity. The proposed DNN from Baghel (BaghelDNN) also compared with our work. Finally, the last SOTA method to compare is from Mei which is based on Wavelet Scattering Transform (WST). All parameters used in this experiment were taken based on their best results from each method.

We tested all SOTA using Yaseen public dataset and evaluated their accuracy performance. It can be seen from Table 5 that our proposed work using ensemble feature surpassed all related SOTAs with 98.3 % accuracy. In the second position is achieved by Yadav feature using KNN classifier followed by Yaseen DNN, Yadav RF, and Baghel DNN.

4.8 Discussion

This research studies about fetal heartbeat sound classification using nine different algorithms. Several studies for pregnant women conducted in developing countries triggered much attention. Research using USG as a medium of recording the sound from fetal has been conducted. The results were astounding. The data contains many clinical histories of maternal, for example, high blood tension, bipolar, abortion, amniotic fluid, admitted abnormal fetus, preeclampsia, and hypoglycemia. All of these conditions affect the heartbeat rhythm. By this condition, we conducted research for arrhythmia detection based on FHB phonograph audio. Due to the data imbalance between normal and abnormal, we utilize three balancing techniques. In audio classification, we extracted the features using MFCC, Chroma, Spectral, and roll-off, then assembled them. The assembly process leads to a large size of the feature. Thus, we implement a dimensionality reduction method, PCA, to improve performance and efficiency. PCA successfully cut down the feature to half of its original size. Even though the size was reduced, the classification performance was remarkable.

To prove the effectiveness of our method we also compared with related SOTAs. As a fair comparison, we tested using the same dataset from Yaseen. Each SOTA parameters were picked based on their top results. Finally, our method achieved the best result.

5. Conclusion

In this paper, we exhibit a phonocardiograph-based audio classification of fetal heartbeats using neural network optimization. The experiment results show significant improvement by utilizing dimensionality reduction and ensemble features. Furthermore, we investigated the data using downsampling, oversampling, and SMOTE. The improvement of classification performance is significantly rising by 0.5 % between downsampling and SMOTE. At the same time, our proposed model overwhelmed the baseline model by 1 % on average. Finally, the best result is achieved with SMOTE-PCA combination with the accuracy and F1-score at 91.7 % and 91.3 %, respectively.

Of course, there is still plenty of room for improvement in future works. For example, the dataset coverage must be improved in quantity and variety. The improvement can be conveyed in a larger population, especially in developing countries with a considerable malnutrition risk. The classification performance can be expanded into more than two classes and be focused on abnormal results. In dimensional reduction, many techniques must be explored to determine the effect on machine learning performance.

Conflicts of Interest

The authors declare no conflict of interest.

Author Contributions

The 1st author contributed in conceptualization, methodology, writing, and data curation. The 2nd author took part in data analysis, software, and data evaluation. The 3rd author contributed in writing review and conceptualization. The 4th author contributed as supervision, conceptualization, and writing review.

Acknowledgments

This research is supported by Pusat Unggulan IPTEK (PUI) Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia.

References

[1] C. Lowe, M. Kelly, H. Sarma, A. Richardson, J. Kurscheid, B. Laksono, S. Amaral, D. Stewart, and D. Gray, “The double burden of malnutrition and dietary patterns in rural Central Java, Indonesia”, *Lancet Reg Health West Pac*, Vol. 14, 2021, doi: 10.1016/j.lanwpc.2021.100205.

[2] V. L. Sigurðardóttir, J. Gamble, B. Guðmundsdóttir, H. Sveinsdóttir, and H. Gottfreðsdóttir, “Reviewing birth experience following a high-risk pregnancy: A feasibility study”, *Midwifery*, Vol. 116, p. 103508, 2023, doi: 10.1016/j.midw.2022.103508.

[3] M. A. Dwitama, Masni, R. Nur, A. Indarty, M. Tahir, A. Mallongi, M. Basir, Mahfudz, and A. Ansyari, “Mapping of high-risk detection of women pregnancy on antenatal care in Talise Health Center, Palu City, Indonesia”, *Gac Sanit*, Vol. 35, pp. S152-S158, 2021, doi: 10.1016/j.gaceta.2021.10.015.

[4] D. E. Effendi, L. Handayani, A. P. Nugroho, and I. Hariastuti, “Adolescent pregnancy preventation in rural Indonesia: a participatory action research”, *Rural Remote Health*, Vol. 21, No. 3, pp. 1-12, 2021, doi: 10.22605/RRH6639.

[5] W. Rahmawati, J. C. Willcox, P. V. D. Pligt, and A. Worsley, “Nutrition information-seeking behaviour of Indonesian pregnant women”, *Midwifery*, Vol. 100, 2021, doi: 10.1016/j.midw.2021.103040.

[6] D. Darmawati, T. N. Siregar, K. Hajjul, and T. Teuku, “Exploring Indonesian mothers perspectives on anemia during pregnancy: A qualitative approach”, *Enfermería Clínica (English Edition)*, Vol. 32, pp. S31-S37, 2022, doi: 10.1016/j.enfcle.2020.11.007.

[7] A. Hazfiarini, R. I. Zahroh, S. Akter, C. S. E. Homer, and M. A. Bohren, “Indonesian midwives perspectives on changes in the provision of maternity care during the COVID-19 pandemic: A qualitative study”, *Midwifery*, Vol. 108, 2022, doi: 10.1016/j.midw.2022.103291.

[8] R. Scott, N. Oliver, M. Thomas, and R. A. Jaffar, “Pregnancy and contraception in women with Pre-Gestational diabetes in secondary Care- A questionnaire study”, *Diabetes Res Clin Pract*, Vol. 182, 2021, doi: 10.1016/j.diabres.2021.109124.

[9] M. Samieinasab and R. Sameni, “Fetal phonocardiogram extraction using single channel blind source separation”, In: *Proc. of 2015 23rd Iranian Conference on Electrical Engineering*, pp. 78-83, 2015, doi: 10.1109/IranianCEE.2015.7146186.

[10] I. Vican, G. Krekovic, and K. Jambrosic, “Can empirical mode decomposition improve

heartbeat detection in fetal phonocardiography signals?", *Comput Methods Programs Biomed*, Vol. 203, 2021, doi: 10.1016/j.cmpb.2021.106038.

[11] R. Kahankova, J. Kolarik, R. Martinek, and A. Durikova, "A Comparative Analysis of Fetal Phonocardiograph Acoustical Performance", *IFAC-PapersOnLine*, Vol. 52, No. 27, pp. 514-519, 2019, doi: 10.1016/j.ifacol.2019.12.715.

[12] S. Gobillot, J. F. Jallon, V. Equy, B. Rivet, P. Y. Gumery, and P. Hoffmann, "Non-invasive fetal monitoring using electrocardiography and phonocardiography: A preliminary study", *J Gynecol Obstet Hum Reprod*, Vol. 47, No. 9, pp. 455-459, 2018, doi: 10.1016/j.jogoh.2018.08.009.

[13] S. L. Oh, V. Jahmunah, C. P. Ooi, R. S. Tan, E. J. Ciaccio, T. Yamakawa, M. Tanabe, M. Kobayashi, and U. R. Acharya, "Classification of heart sound signals using a novel deep WaveNet model", *Comput Methods Programs Biomed*, Vol. 196, 2020, doi: 10.1016/j.cmpb.2020.105604.

[14] N. Mei, H. Wang, Y. Zhang, F. Liu, X. Jiang, and S. Wei, "Classification of heart sounds based on quality assessment and wavelet scattering transform", *Comput Biol Med*, Vol. 137, 2021, doi: 10.1016/j.compbiomed.2021. 104814.

[15] Yaseen, G. Y. Son, and S. Kwon, "Classification of heart sound signal using multiple features", *Applied Sciences (Switzerland)*, Vol. 8, No. 12, 2018, doi: 10.3390/app8122344.

[16] A. Raza, "Factors associated with vulnerability of patients to medical errors", *Int J Med Sci Public Health*, No. 0, p. 1, 2019, doi: 10.5455/ijmsph.2019.0512522052019.

[17] A. Yadav, A. Singh, M. K. Dutta, and C. M. Travieso, "Machine learning-based classification of cardiac diseases from PCG recorded heart sounds", *Neural Comput Appl*, Vol. 32, No. 24, pp. 17843-17856, 2020, doi: 10.1007/s00521-019-04547-5.

[18] S. K. Ghosh, R. N. Ponnalagu, R. K. Tripathy, and U. R. Acharya, "Automated detection of heart valve diseases using chirplet transform and multiclass composite classifier with PCG signals", *Comput Biol Med*, Vol. 118, 2020, doi: 10.1016/j.compbiomed.2020.103632.

[19] B. Bozkurt, I. Germanakis, and Y. Stylianou, "A study of time-frequency features for CNN-based automatic heart sound classification for pathology detection", *Comput Biol Med*, Vol. 100, pp. 132-143, 2018, doi: 10.1016/j.compbiomed.2018.06.026.

[20] N. Baghel, M. K. Dutta, and R. Burget, "Automatic diagnosis of multiple cardiac diseases from PCG signals using conVolutional neural network", *Comput Methods Programs Biomed*, Vol. 197, 2020, doi: 10.1016/j.cmpb.2020.105750.

[21] C. Liu, D. Springer, Q. Li, B. Moody, R. A. Juan, F. J. Chorro, F. Castells, J. M. Roig, I. Silva, A. E. W. Johnson, Z. Syed, S. E. Schmidt, C. D. Papadaniil, L. Hadjileontiadis, H. Naseri, A. Moukadem, A. Dieterlen, C. Brandt, H. Tang, M. Samieinasab, M. R. Samieinasab, R. Sameni, R. G. Mark, and G. D. Clifford, "An open access database for the evaluation of heart sound algorithms", *Physiol Meas*, Vol. 37, No. 12, pp. 2181-2213, 2016, doi: 10.1088/0967-3334/37/12/2181.