[1] **Tarang Pande**

Detection of Heart Sound Abnormality using Artificial Intelligence & Machine Learning

Journal of Electrical Systems

Abstract: - Phonocardiogram signals – or PCG signals – are the recordings of sounds and murmurs created by the heart. They are used to detect heart abnormalities in a clinical environment. In this paper, we try to automate the detection process. We look at heart signal recordings and analyze them using advanced machine learning algorithms and models. These algorithms are capable of identifying subtle patterns and anomalies in the heart signal data that may be indicative of underlying cardiac conditions. By automating the process, we can process large volumes of data in a fraction of the time it would traditionally take to do the same. This not only increases the efficiency of our analysis but also allows us to deliver results in real-time, which could be critical in urgent care situations. Also, incorporating machine learning models into the detection mechanism allows the machine to continue developing by looking at more samples and become better at predicting abnormalities. It will also allow the model to incorporate any new heart abnormalities in the future without much difficulty.

Keywords: heart sound abnormality, signal processing, abnormal heart sounds, systole, diastole, machine learning.

I. INTRODUCTION

Abnormal heart sounds are murmurs that can be described as a whooshing or blowing sound which are caused by an unstable blood flow. These abnormal sounds are heard with the use of a stethoscope in a clinic. There are different types of murmurs: one is harmless while the other ones are very harmful and dangerous. Fainting, blood clots, strokes, heart attack, cardiac arrest, or complete heart failure are some of the implications of a dangerous abnormal heart murmur (Grzegorczyk, et al., 2016). Phonocardiogram (PCG) signals are used for heart disease detection. They contain auditory data regarding the operation of the heart. They mainly comprise four different states: First heart sound (S1), systole, second heart sound (S2) and diastole (Potes, Parvaneh, Rahman, & Conroy, 2016). These parts tell us about the condition of the heart. When the signals are normal, only the four main parts are present but when they are abnormal, there are other sounds like S3, S4 and murmurs (Homsi, et al., 2016). There are often extra noises such as outside noises and the sound of other processes inside the body that make it hard to obtain an accurate reading of the signal.

Despite the challenges, the combination of signal processing and machine learning (ML) holds great promise for the automatic detection of abnormal sounds in PCG signals. By using advanced algorithms and learning through data, ML-based models can give near- perfect results if done right. The main advantage of using ML-based models is their ability to learn and adapt to large amounts of data. As the model is exposed to diverse PCG signals, it can learn the complex patterns and variations associated with different types of abnormal sounds. This enables the system to generalize its knowledge and make accurate predictions on other PCG signals. The integration of ML allows for the continuous improvement and refinement of the model as new diseases related to the heart are discovered.

II. DATA COLLECTION

We acquired the heart recordings that were used in this study from the Physionet data bank to train the machine learning models. The dataset of the recordings contained a collection of 3,126 heart recordings, ranging from 5 seconds to 120 seconds. The recordings were collected from adult patients, including both males and females. Each file contained a PCG signal and there was a separate document labelling each recording as being from a healthy patient or a patient with any heart abnormalities (Liu, et al., 2016).

[1] New Jersey, United States of America. pandet.tp@gmail.com

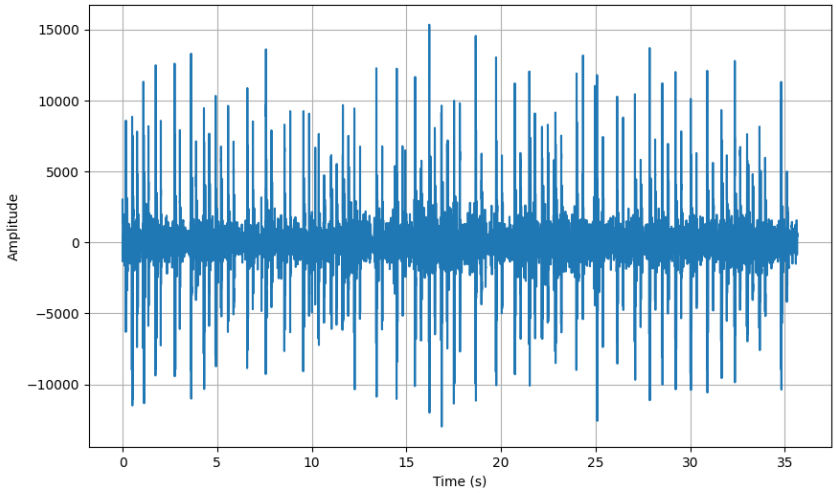

Figure 1. PCG signal for a normal person.

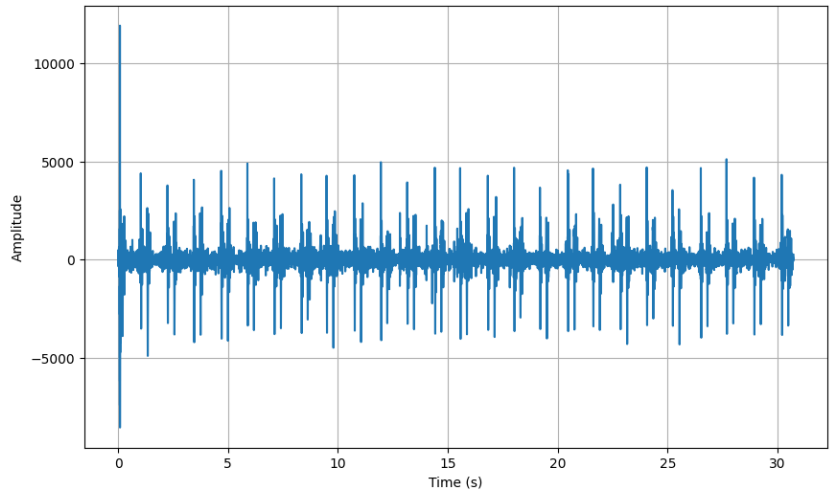

Figure 2. PCG signal for a person with heart abnormalities

III. PRE-PROCESSING

After getting all the signals from the data bank, copies of the signals were made. The first copy of signals was re-sampled to 1000Hz using Butterworth high-pass and Polyphase anti-aliasing filter. The Butterworth high-pass filter filters the data based on the given cutoff frequency, sampling frequency, and filter order. The filtered data is returned and stored for use later on (Virtanen, et al., 2020). The other copy of the signals is normalized using the standardization formula $Z = \frac{X - mean(X)}{std(X)}$, where X is the signal. The normalized data is used for getting the time domain features and the filtered data is used for getting the frequency domain features and the heart rate. A time domain feature is found from analyzing signals or mathematical function, in reference to time. A frequency domain is the same thing but it is in reference to frequency. The figure below references a normalized signal (Hunter, 2007).

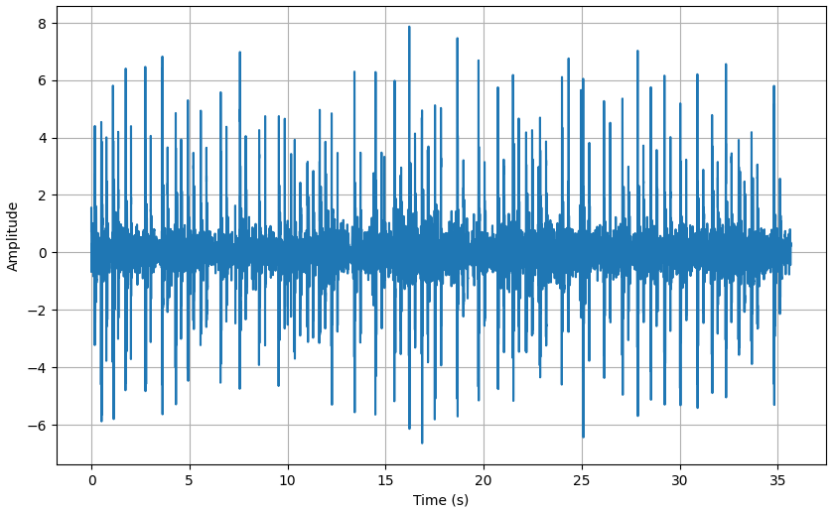

Figure 3. Normalized PCG signal for a normal person

IV. Feature Extraction

After filtering and normalizing the signals, we extract features from them that will help us in classifying the signals as normal or abnormal, and make a boxplot out the extracted features to compare the two types (pandas development team, 2020).

Each feature's function can be found below:

1. 'mean' (Time domain) refers to the average value of the signal (Harris, et al., 2020).
2. 'min' (Time domain) refers to the minimum value of the signal (Harris, et al., 2020)
3. 'max' (Time domain) refers to the maximum value of the signal (Harris, et al., 2020).
4. 'skewness' (Time domain) is the measure of the asymmetry of the probability distribution of the signal (Virtanen, et al., 2020).
5. 'kurtosis' (Time domain) represents whether the signal distribution is flat or peaky.
6. 'psd_mean' (Frequency domain) represents the average value of the power spectral density (PSD) of the signal (Virtanen, et al., 2020).
7. 'psd_sum_amplitude' (Frequency domain) represents the sum of the amplitudes in the power spectral density (PSD) of the signal (Harris, et al., 2020).
8. 'psd_min' (Frequency domain) represents the minimum value of the power spectral density (PSD) of the signal (Harris, et al., 2020).
9. 'highest_amplitude_freq' (Frequency domain) represents the frequency corresponding to the highest amplitude in the signal's power spectrum (Harris, et al., 2020).
10. 'psd_max' (Frequency domain) represents the maximum value of the power spectral density (PSD) of the signal. (Virtanen, et al., 2020).
11. 'spectral_centroid' (Frequency domain) is the measure of the center of mass of the signal's spectrum (Virtanen, et al., 2020).
12. 'heart_rate' (Frequency domain) refers to the number of times a person's heart beats per minute in the signal (Harris, et al., 2020).

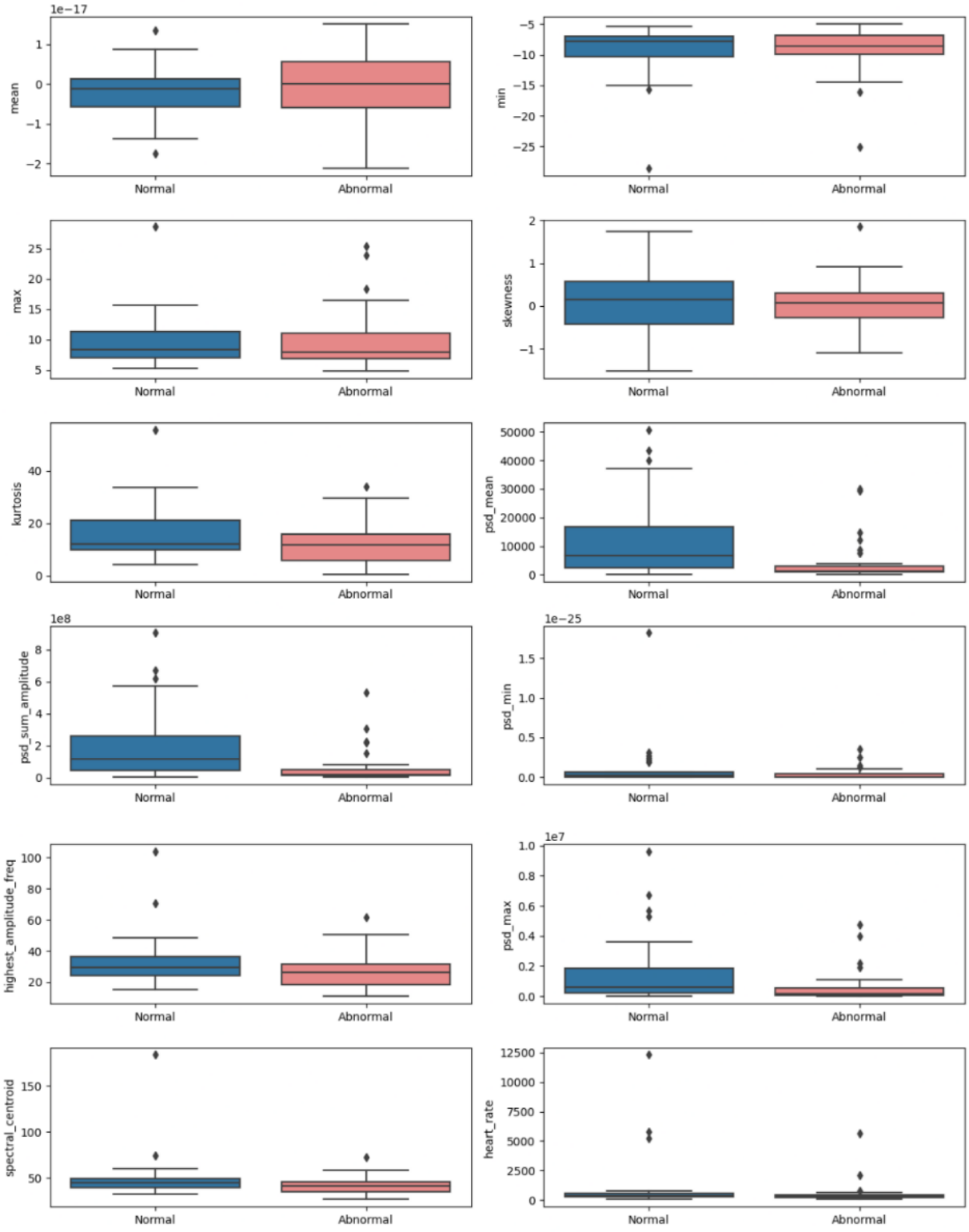

Figure 4. Box plots for the features separated by class

V. MACHINE LEARNING PHASE

Once the features are extracted, each of the 12 features have been created into arrays. The arrays are passed through the 'train test split' function provided by sklearn to get the training and validation sets setup the features which will be used by the machine learning model to distinguish between normal and abnormal signals. The training set is used to train the machine learning models, which help the model learn patterns in the data. The sample size for validation is 20% at a time and there is randomization to ensure that all samples are taken. The training and

validation set for the samples are scaled using the function MinMaxScaler provided by sklearn to ensure that all features are on a similar scale to improve the performance and have consistency (Pedregosa, et al., 2011).

Machine learning models, such as LogisticRegression (LR), DecisionTreeClassifier (CART), KNeighborsClassifier (KNN), GaussianNB (NB), SVC (SVM) and LinearDiscriminantAnalysis (LDA) are chosen (Vernekar, Nair, Vijaysenan, & Ranjan, 2016). Stratified k-fold cross-validation is done with three splits and the shuffling of the data. This technique is used as there is only a small number of samples and the abnormal class is somewhat underrepresented. It is also used to estimate the accuracy and precision of the machine learning models. The cross-validation results are then recorded for each model. The models were then also assessed on separate validation sets to ensure accuracy. Each model was trained on all of the features and then the models were tested. The trained models were employed to make prediction on the samples as either being 0 (normal) or 1(abnormal). A confusion matrix was employed to check the performance. The number of true negatives (tn), false positives (fp), false negatives (fn), and true positives (tp) were extracted from the validation set which allowed us to find the sensitivity and specificity (Pedregosa, et al., 2011).

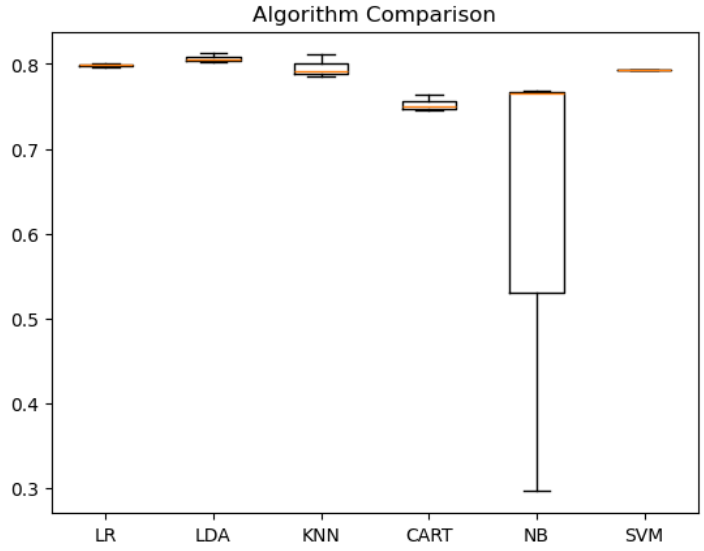

Figure 5. ML model comparison

After doing the model comparisons, we the use the LDA (LinearDiscriminantAnalysis) model for the final model as it gives the highest accuracy and then train it using the extracted features. The trained model is then used to employ predictions on the samples as either normal or abnormal later in the results section.

VI. RESULTS

After getting the predictions, we make a classification report to compare the labels of the samples and the predictions that we got from the model. We get an accuracy of 0.79 or 79%. We also find the precision (how accurate the model is in its prediction), recall (how often the model predicted correctly), and the f1-score (a measure that indicates the model's predictive performance calculated from tp, tn, fp, and fn) (Pedregosa, et al., 2011).

Table 1. Precision, F1-Score, Recall, and Accuracy numbers

	Precision	**Recall**	**F1-Score**
Class 0	0.85	0.80	0.82
Class 1	0.93	0.78	0.85
Accuracy			0.79
Macro Avg	0.72	0.69	0.84
Weighted Avg	0.72	0.69	0.84

VII. Discussion

As we see from the results, the accuracy comes to be 79%. It could be improved with a better understanding of the causes of heart abnormalities and more clinical knowledge of the heart to fine-tune the model. The sample size, number of features, and the limited models used also held back the results. There is noise in some of the signals that was not able to be removed due to time constraints, which also hampers the final results. Using more samples than 3000, testing more features, and implementing other machine learning models used in this study would improve the accuracy.

VIII. Conclusion

This study is to show how machine learning can be used to automate the detection of any abnormalities in the heart and is not a heavy clinical study. Conducting this study in a professional environment will improve it. Still, the results (79% accurate in detecting if the signal is abnormal) are remarkable. These results illustrate the potential of using machine learning and signal processing in the medical field. If perfected, this method will allow the detection of any heart conditions swiftly and with great accuracy while also being more accessible than the other ways of diagnoses.

Acknowledgements

I would like to thank the Physionet Databank to provide me with the heart sound signals used in this research paper.

References

[1] Grzegorczyk, I., Soliński, M., Łepek, M., Perka, A., Rosiński, J., Rymko, J., . . . Gierałtowski, J. (2016). PCG classification using a neural network approach. *2016 computing in cardiology conference (CinC)* (pp. 1129–1132). IEEE. Retrieved from https://ieeexplore.ieee.org/abstract/document/7868946

[2] Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., . . . Oliphant, T. E. (2020, September). Array programming with NumPy. *Nature, 585*, 357–362. doi:10.1038/s41586-020-2649-2

[3] Homsi, M. N., Medina, N., Hernandez, M., Quintero, N., Perpiñan, G., Quintana, A., & Warrick, P. (2016). Automatic heart sound recording classification using a nested set of ensemble algorithms. *2016 Computing in Cardiology Conference (CinC)* (pp. 817–820). IEEE. Retrieved from https://ieeexplore.ieee.org/abstract/document/7868868

[4] Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. *Computing in Science & Engineering, 9*, 90–95. doi:10.1109/MCSE.2007.55

[5] Liu, C., Springer, D., Moody, B., Silva, I., Johnson, A., Samieinasab, M., . . . Clifford, G. D. (2016). Classification of heart sound recordings-the physionet computing in cardiology challenge 2016. *PhysioNet*. Retrieved from https://www.physionet.org/content/challenge-2016/1.0.0/papers/

[6] pandas development team, T. (2020, February). pandas-dev/pandas: Pandas. *pandas-dev/pandas: Pandas*. Zenodo. doi:10.5281/zenodo.3509134

[7] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., . . . Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research, 12*, 2825–2830. Retrieved from https://www.jmlr.org/papers/volume12/pedregosa11a/pedregosa11a.pdf

[8] Potes, C., Parvaneh, S., Rahman, A., & Conroy, B. (2016). Ensemble of feature-based and deep learning-based classifiers for detection of abnormal heart sounds. *2016 computing in cardiology conference (CinC)* (pp. 621–624). IEEE. Retrieved from https://ieeexplore.ieee.org/abstract/document/7868819

[9] Vernekar, S., Nair, S., Vijaysenan, D., & Ranjan, R. (2016). A novel approach for classification of normal/abnormal phonocardiogram recordings using temporal signal analysis and machine learning. *2016 computing in cardiology conference (CinC)* (pp. 1141–1144). IEEE. Retrieved from https://ieeexplore.ieee.org/abstract/document/7868949

[10] Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., . . . Contributors, S. 1. (2020). SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods, 17*, 261–272. doi:10.1038/s41592-019-0686-2