Precision/Recall Trade-Off Analysis in Abnormal/Normal Heart Sound Classification

Jeevith Bopaiah[2] and Ramakanth Kavuluru[1,2]*

[1] Division of Biomedical Informatics, Department of Internal Medicine
[2] Department of Computer Science
University of Kentucky, Lexington, KY
{jeevith.bopaiah,ramakanth.kavuluru}@uky.edu

Abstract. Heart sound analysis is a preliminary procedure performed by a physician and involves examining the heart beats to detect the symptoms of cardiovascular diseases (CVDs). With recent developments in clinical science and the availability of devices to capture heart beats, researchers are now exploring the possibility of a machine assisted heart sound analysis system that can augment the clinical expertise of the physician in early detection of CVD. In this paper, we study the application of machine learning algorithms in classifying abnormal/normal heart sounds based on the short ($\leq$ 120 seconds) audio phonocardiogram (PCG) recordings. To this end, we use the largest public audio PCG dataset released as part of the *2016 PhysioNet/Cardiology in Computing Challenge*. The data comes from different patients, most of who have had no previous history of cardiac disease and some with known cardiac diseases. In our study, we use these audio recordings to train three different classification algorithms and discuss the effects of class imbalance (normal vs. abnormal) on the precision-recall trade-off of the prediction task. Specifically, our goal is to find a suitable model that takes into account the inherent imbalance and optimize the precision-recall trade-off with a higher emphasis on increasing recall. Bagged random forest models with majority (normal) class under sampling gave us the best configuration resulting in average recall over 91% with nearly 64% average precision.

1 Introduction

For the past decade, cardiovascular diseases (CVD) have been the leading cause of deaths around the globe. According to the WHO statistics, as of 2015, ischemic heart disease is the "world's biggest killer" (`http://www.who.int/mediacentre/factsheets/fs310/en/`). According to a report published in 2017 by American Heart Association, CVD accounts for 801,000 deaths in the United States [1]. Most of these deaths could be prevented if the diseases were detected in their early stage. Auscultation is a procedure used by the physicians to examine the heart. It involves listening to the heart sounds to detect abnormality in the heart. This requires substantial experience and is a complex process prone

* corresponding author

to human error. Also, the patient to doctor ratios are extremely high in certain parts of the world (up to tens of thousands) and hence manual examination is not ideal in many cases. Given these situations, cloud based solutions that allow more accurate preliminary examination of heart health based on heart sounds may offer an important alternative. Central to such a service would be a high quality predictive model that can identify abnormal heart sounds automatically. To make this a reality, researchers around the world are building expert annotated datasets and machine learned models. The *2016 PhysioNet/Computing in Cardiology Challenge (CinC)* [5] provided the largest public heart sound database with which researchers built supervised models and tested against a hidden test set. Although the competition ended late 2016, the hidden test set has not been made public yet. In this paper, we study the efficacy of classical machine learning algorithms in identifying abnormal heart sounds with a focus on the precision-recall trade-off. Before we proceed, we first discuss how heart sounds are generated and measured.

Heart sounds are produced by four distinct events that take place in the heart. These four events correspond to the mechanical activity of opening and closing of the valves in the heart. Each heart beat is triggered by an electrical impulse inside the heart that causes the atrium and ventricles to contract and relax alternatively [4]. This consecutive contraction and relaxation event draws impure blood into the heart and pumps out pure blood to the rest of the body. Each heart cycle is composed of these four events that occur in quick succession in a particular order. The actual sequence of events is *S1, systole, S2*, and *diastole* where S1 and S2 correspond to the fundamental sounds made by the heart via its mechanical movements. Along with these, the heart recordings may also contain other sounds such as systolic ejection click (EC), mid-systolic click (MC), the diastolic sound (OS), as well as heart murmurs caused by the flow of blood [8]. All these sounds can be captured using a phonocardiograph which produces an audio file. The audio recording should at least be long enough to contain an entire heart cycle. In this project our task involves developing a predictive model that analyzes the sound patterns of the audio file to predict the corresponding heart beat as either normal or abnormal. This allows more accessible, real time monitoring of the heart that can be used to assist physicians in preliminary checks for CVDs.

Given this motivation, researchers have been working in the field of heart sound analysis for the past five decades but most of their efforts have had drawbacks in terms of access to very few heart sound recordings, lack of a separate test dataset, and failure to use a variety of PCG recordings [5]. However, these drawbacks have been mitigated with the introduction of the *2016 PhysioNet/CinC Challenge* dataset. Some of the recent works [11, 12, 17] on this dataset include the use of deep neural networks and ensemble approaches (more details in Section 8). However, most of these efforts do not analyze the trade-off between recall and precision. Actually, they all analyze accuracy which is defined for them as the simple mean of recall and specificity (which is different from precision). However, for classification tasks with imbalanced datasets where the minority class is the positive class that is of interest, it is well known that precision and recall are more important [13]. Our effort is focused on precision-recall analysis while also disclosing accuracy information.

2 Dataset

The dataset [6, 8] used in our experiments was obtained from a publicly available heart sound database which was hosted by the *PhysioNet* group. This dataset was compiled by various researchers around the world who have collected eight heart sound databases, each sourced from different healthcare facilities and home visits. These heart sounds were recorded with a sampling rate of 44 kHz which was then downsampled to 2000 Hz. Out of these eight databases, six were made publicly available for training the models while the remaining two databases along with few records from training dataset were kept private as blind test data. The summary of the training dataset is shown in Table 1.

Table 1: Physionet/CinC Challenge training dataset summary [8]

Database Name	# Raw Recordings		
	Abnormal	Normal	Total
Database-a	292	117	409
Database-b	104	386	490
Database-c	24	7	31
Database-d	28	27	55
Database-e	183	1958	2141
Database-f	34	80	114
Total	665	2575	3240

The public dataset consists a total of 3,240 heart sound recordings. The length of each recording varies between 5 and 120 seconds. The average length of heart cycle within each recording is 1.5–2 seconds. Thus, each recording is long enough to contain more than one heart cycles. Typically, higher number of heart cycles present in a recording allows for a better representation of abnormal patterns during feature extraction. This is analogous to the fact that learning algorithms generalize better with more relevant data points. The entire dataset has around 80,000 heart cycles in it. These recordings can be parsed to produce a vector of amplitudes varying in time.

3 Methods

The architecture of our predictive model for the heart sound classification task is shown in Figure 1. In this approach, we experiment with three well known classification algorithms: random forests, logistic regression, and support vector machines (SVM). Of the three algorithms, we found random forest algorithm to be more effective in classifying the heart beats as either normal or abnormal if F-score is the chosen measure[1]. Random forests are an ensemble model formed by employing multiple decision trees as base classifiers. Each tree in the random forest is trained on a randomized smaller subset of the full set of features. These individual trees behave as weak learners with complementary characteristics and hence are combined to create a powerful learning algorithm that uses a voting mechanism among the different trees to obtain the final predictions. Furthermore, we have optimized the hyper-parameters: *number of trees* and *depth of each tree* by using an exhaustive search algorithm. We found the best configuration involved 200 trees with a tree depth of up to 20 levels.

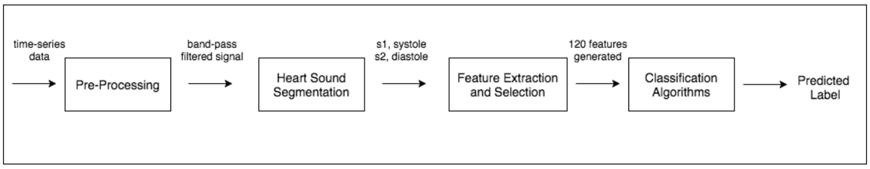

Fig. 1: Predictive modeling pipeline used in heart sound classification

The different stages involved in our predictive model are as follows:

3.1 Pre-processing

At the time of recording the heart sound, noises from the external environment or internal body functions are recorded along with the actual heart sounds. These background noises distort the actual signal and have a negative influence on the final predictions. The pre-processing stage involves de-noising the signal to contain only the actual heart sounds. The input signal, in the form of time series values of amplitudes, is further downsampled to 1000 Hz [11]. Downsampling is a process in which we reduce the number of data points/second in the input signal. The pre-processed signal now consists of 1000 amplitude values per second. This is useful especially when the sampling rate is much larger than the highest frequency component of the signal and processing the data becomes a challenge. From literature [8], we know that the heart sounds lie in the frequency range of 25 Hz – 500 Hz. According to the Nyquist sampling theorem [3], there is no information loss if the sampling rate is at least twice the highest frequency component of the signal. Since we know that the highest desired frequency component is 500 Hz, we can downsample the signal to 1000 Hz without much loss of

[1] Henceforth, we only discuss the results using the random forest approach. Comparisons with the other two classifiers are presented in Section 6

information. In the next step, we pass the signal through band-pass filters module that retains the frequencies in the range 25 Hz-500 Hz and eliminates the rest. This helps to weed out undesired frequencies less than 25 Hz and greater than 500 Hz. The signal is passed through a spike removal process that removes sharp peaks. The spike removal process helps in removing the noises from the external environment that appear as spikes in the signal. Finally, the signal is normalized to reduce the effect of extremely large or small amplitudes.

3.2 Heart Sound Segmentation

In this stage, each heart sound recording is segmented into four distinct heart sounds: S1, systole, S2, diastole. Each of the four heart sounds exhibits a distinct waveform pattern. Any variation in one or more of these sounds could potentially indicate an abnormality. Segmenting the entire heart sound recording helps in analyzing each of these four heart sounds for abnormal patterns. As suggested by the organizers of the 2016 PhysioNet challenge, we used the available state of the art segmentation algorithm developed by Springer et al. [14]. They use ECG as a reference signal to identify the approximate locations of the four heart sounds on a PCG signal. The PCG signal corresponds to the actual heart sound recording. Figure 2 shows the segmented PCG signal along with the ECG waveform.

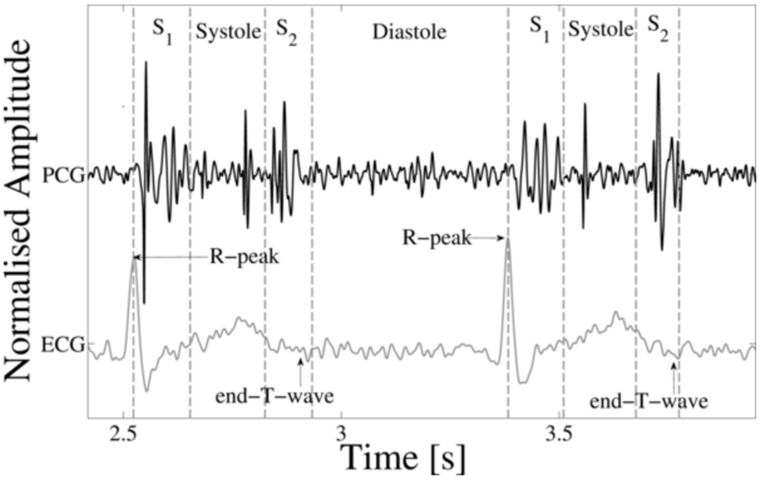

Fig. 2: PCG and ECG waveforms (from Liu et al. [8])

ECG measures the electrical impulse of the heart and is less prone to noise than a PCG signal. However, recording ECG is an expensive process and it is recommended by the physician only if needed at a later stage. The R-peaks (shown in Figure 2) of the ECG coincide with the S1 phase of the heart beat. Similarly, the end-T-wave of the ECG coincides with the end of the S2 phase. Thus, using the R-peaks and end-T-waves of the ECG, the location of the heart sounds on a PCG are identified. The segmentation algorithm uses logistic regression coupled with a hidden semi-Markov model to predict the most likely sequence of states for each recording. The hidden semi-Markov model maximizes the likelihood of each data point to be in one of the four states while the logistic regression classifier models the expected duration densities for each state.

3.3 Feature Extraction and Selection

Based on the boundary regions of S1 identified in the segmentation step, we divide the entire heart sound recording into individual heart cycles. The features are extracted from each heart cycle and then averaged across the other heart cycles in the recording[2]. The features extracted from each heart cycle can be classified into two feature classes: time domain features and frequency domain features. The time domain features are comprised of the aggregate measures of the heart sound states. They can be further categorized into PCG intervals and PCG amplitudes.

The PCG intervals measure the time intervals of the various components of the heart recording. Features from the PCG intervals include mean and standard deviation of the following:

1. Length of the heart cycle
2. S1 interval length
3. Systole interval length
4. S2 interval length
5. Diastole interval length
6. Ratio of length of the systolic interval to the length of the heart cycle
7. Ratio of length of the diastolic interval to length of the heart cycle
8. Ratio of length of the systolic interval to that of the diastolic interval

The PCG amplitudes measure the aggregates of the amplitude values in the signal. These include the mean and standard deviation of the following:

1. Ratio of the mean amplitude in systole to the mean amplitude in S1
2. Ratio of the mean amplitude in diastole to the mean amplitude in S2
3. Skewness and kurtosis of amplitude in S1
4. Skewness and kurtosis of amplitude in systole
5. Skewness and kurtosis of amplitude in S2
6. Skewness and kurtosis of amplitude in diastole

Thus 36 features have been extracted from the time domain signal. The remaining 84 features were obtained from the frequency domain signal by using acoustic properties of the sound waves [11]. These include

1. Power Spectral Density
2. Mel-Frequency Cepstral Coefficients.

[2] We emphasize that prediction of abnormality is made per recording, not per cycle, given a full recording's multiple cycles together provide the signal for prediction

Power Spectral Density (PSD) [9, Chapter 11]: It refers to the variances in amplitude in terms of the frequency of the signal. In simple terms, it measures the distribution of energy over the various frequency components of the signal. In order to compute the PSD, we first extract the frequency components that exists in a signal. Thus, the input time domain signal needs to be transformed into frequency domain signal. Fast Fourier transform is a signal processing technique that converts a time domain signal to its frequency domain. The PSD is measured for each of the four heart sounds: S1, systole, S2, diastole across nine different frequency bands: 25-45 Hz, 45-65 Hz, 65-85 Hz, 85-105 Hz, 105-125 Hz, 125-150 Hz, 150-200 Hz, 200-300 Hz, 300-400 Hz. This gives us a vector of 9 values for each of the four types of sounds and a total of 36 features for each heart cycle.

Mel-Frequency Cepstral Coefficients (MFCC) MFCC [7] is a powerful transformation technique that is popular among the speech recognition enthusiasts. It is based on the premise that humans perceive sound on a non-linear scale. In other words, the relationship between energy present in the sound and the loudness perceived by the human ear is non-linear as we transition from lower frequencies to higher frequencies. Increasing the intensity of a sound by a factor X, does not increase the loudness we hear by the same factor X. This is especially true for higher frequencies, where two sounds of frequencies, say for example, 4000 Hz and 4500 Hz are indistinguishable to the human ear. This non-linear relationship between the perception of the sound versus the actual energy present in the sound is modeled on a scale known as the mel scale. MFCC is an extension of the power spectral density graph in which the frequency in hertz is converted into frequency in mel using the below formula [7]

$$mel(f) = 2595 * \log_{10}\left(1 + \frac{f}{700}\right).$$

These frequencies in mels are used to create triangular filters that capture the energy present within each filter. Discrete cosine transformations are applied to the energies obtained from the mel filters to obtain the MFCC. The number of coefficients correspond to the number of filters. We have used 12 filters for each heart sound and each coefficient is considered as a feature. Thus, we have a total of $12 \times 4 = 48$ features.

Combining all the features we now have 120 features that can be used in training the random forest classifier. For feature selection, we used random forest classifier to identify a set of 81 informative features that determine the classification of the heart sound. Feature importance in random forests is determined by ranking the features based on its information gain. In each of the constituent decision trees, the feature chosen at each node is the one that maximizes the information gain at that node. Thus based on the ‘gini impurity’ (GI) measure, the features that maximizes the information gain across the different decision trees are ranked higher in the feature selection list [2]. We use GI because it is obtained as a direct consequence of using the random forest classifier and closely relates to the classifier’s underlying principle. Specifically, a feature with low GI score is more desirable to the classifier than a feature with high GI score. Since random forest classifier also takes into consideration the node impurities while predicting the label, GI appears to be a more appropriate feature selection criterion. The feature importance scores are normalized across all the features. By experimenting with thresholds of 0.006, 0.005, 0.004, 0.003 on feature importances, it was observed that a threshold of 0.005 resulted in 81 features that produced the best results. Among the 81 features, we found 2 feature classes (shown in Table 2) that were more prominent than the rest.

Table 2: Prominent features of the random forest classifier

Feature Class	Feature Score
Mel Frequency Cepstral Coefficients of Diastole Region	0.069
Mel Frequency Cepstral Coefficients of Systole Region	0.027

3.4 Random Forest Classifier Configurations

The 81 features obtained from feature selection process were used in training a random forest classifier. Initially, we used all the 3240 samples in training the classifier and noticed that the recall was averaging around 70%. Since the objective of this classification problem is to maximize recall without making prohibitive compromises on precision, we have implemented different configurations to study the effects of majority class under sampling on recall. These configurations are constructed by retaining all the samples from the minority class and varying the proportions of the majority class. On analyzing the results of these different configurations, we noticed that as the imbalance between the two classes decreased, the recall improves up to a certain threshold, beyond which it results in a loss of precision. In order to demonstrate this effect, we describe the last four configurations that capture the shift from increase in recall to decrease in precision.

To overcome class imbalance, we under sample the majority class and use a bagging approach on different bootstrap samples [15]. The dataset is split into 90% training and 10% test sets with train size of 2925 samples and test size of 315 samples. The test proportions of the positive and negative samples have been retained as in the original dataset (roughly 20% positive and 80% negative). Given we only under sample majority class, the minority class count is always the same (specifically, from Table 1 we have $665 \cdot 0.9 \approx 600$)

- **Model Configuration 1**: In this configuration, the number of positive (abnormal) examples is kept constant at 600 and the negative sample size is varied in increments of 100 (so 600, 700, ..., 2200) for each model. Thus we have 17 different classifiers. For each negative sample size, we train ten

classifiers, for each of which the negative examples are chosen without replacement so that there are no duplicates. The final prediction is based on a voting mechanism with equal contribution from each of the 170 classifiers.
- **Model Configuration 2**: This configuration is a subset of the first configuration. We choose 600 positive examples and 900 negative examples. We train ten models with random sampling on the negative examples. The final prediction is based on the voting with the ten models.
- **Model Configuration 3**: This is similar to configuration 2 except that the number of negative examples is decreased to 800.
- **Model Configuration 4**: This is also similar to the second configuration with the number of negative examples further reduced to 700.

Given there are an even number of classifiers, ties are broken in favor of the minority class.

Evaluation Strategy: The 2016 PhysioNet challenge organizers use recall (also called sensitivity) and specificity metrics where specificity is the ratio of the true negatives to the sum of true negatives and false positives. For this particular task, the proposed recall and specificity metrics depend on specific weights determined by the number of 'noisy' and 'clean' records. Unfortunately, we do not have access to the noisy/clean labels for the public database; they were only provided for the hidden test set that is still not made public. We believe that precision, recall, and F-score are more informative for this task with a minority positive class of interest. For this, a realistic evaluation of the predictive model should account for the prevalence among the two classes. The area under the receiver operating characteristic (ROC) curve, representing a trade-off between recall and specificity, is shown to overestimate the performance of the model in imbalanced datasets with a minority positive class [10]. Hence, we chose the precision/recall as the main metrics that take into account the prevalence of the disease while evaluating the performance of the predictive model.

4 Results and Discussion

The results of the four different configurations are shown in Table 3. A quick glance at the results, especially the F-score and accuracy[3] may appear to be more or less similar in all the four configurations. This is also true to some extent until we take into considerations the key metrics: precision and recall. Though we could maximize any of the evaluation metric listed above, the one more suited for this task is maximizing the recall without incurring prohibitive losses in precision. A more fine grained observation reveals that the recall measure improves as the number of negative samples decreases across the different configurations. In the second configuration, we choose 900 negative samples. This was based on our experiments which showed that the recall drops significantly if the number

[3] The notion of accuracy used here is the same as in the 2016 CinC challenge where it is set to (recall+specificity)/2

of negative samples is beyond 900. Similarly, when the negative samples are reduced to below 700, the drop in precision is greater than the improvement in the recall.

Table 3: Random forest classifier performance measures

	Precision	Recall	F-score	Accuracy
Model Configuration 1	0.778	0.754	0.766	0.849
Model Configuration 2	0.697	0.815	0.752	0.862
Model Configuration 3	0.626	0.877	0.731	0.870
Model Configuration 4	0.615	0.908	0.733	0.879

With configuration 4, we achieve a recall of 0.908 and precision of 0.61. This means that it could catch over 90% of the patients with cardiovascular diseases with precision of 61% – implying 39% of cases predicted as abnormal are actually normal. Even though the numerical value of precision makes the classifier appear very poor, for practical purposes this is not really a major hurdle. Specifically, given the number of instances predicted to be of the minority class is very low compared with the number predicted for the majority class, the manual burden of weeding out these additional healthy cases is also low given the 39% proportion is out of instances predicted to be abnormal.

In order to assess the stability of the results from configuration 4, we repeated the experiment 40 times by considering a different train-test split each time. The average results of the 40 runs are shown in Table 4. These results are similar to those in Table 3. To demonstrate this, we establish confidence intervals on the results obtained from the 40 runs. At 95% confidence, the accuracy is shown to be within 0.888 ± 0.0068 The tight bounds on the accuracy show that the performance is expected to generalize well.

Table 4: Average results of Config 4 via experiments with 40 distinct train-test splits

	Precision	Recall	F-score	Accuracy
Model Configuration 4	0.637	0.912	0.749	0.887

The precision-recall (PR) curves for the four configurations are shown Figures 3–6. As we can see, the area under the PR curves (AUPRC) is similar in all configurations but is slightly lower in the 4th configuration at 0.82, which is around four points lower compared with the first configuration. However, it is also clear (as we conveyed earlier) from a practical perspective, configuration 4 is more desirable.

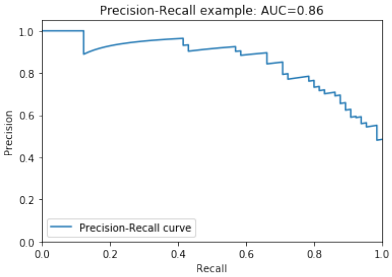

Fig. 3: Model Configuration 1

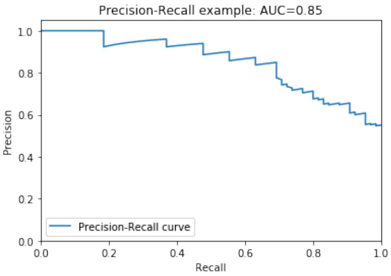

Fig. 5: Model Configuration 2

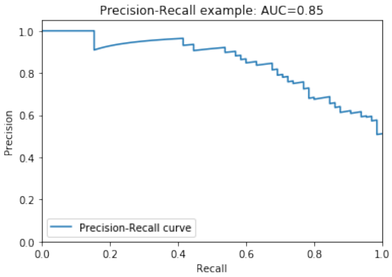

Fig. 4: Model Configuration 3

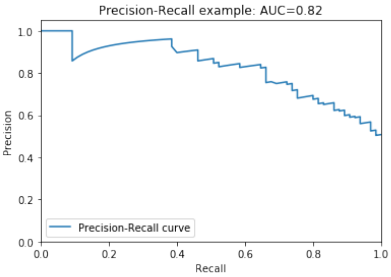

Fig. 6: Model Configuration 4

5 Error Analysis

From the results of the random forest classifier, we know that the model suffers from a low precision score. To analyze the classification errors, we provide our error analysis on one of the 40 runs we conducted to generate results in Table 4. The prediction results, in terms of true positives, true negatives, false positives, and false negatives, are shown as a confusion matrix in Figure 7.

The confusion matrix indicates an error of 14.4% false positives and 9.2% false negatives. On analyzing the euclidean distance between the feature vectors of the training samples and the misclassified test samples, we found that a significant portion of the test instances were closer to their incorrectly predicted class than their true class. Thus feature characteristics caused some of the samples to be misclassified. Specifically, Table 5 shows the percentages of false positive and false negative errors that are similar to positive and negative classes, respectively. 61.1% of test errors that were incorrectly predicted as abnormal, were closer to the abnormal training samples on average. Similarly, 83.33% of test errors that were incorrectly predicted as normal were closer to the normal training samples. It is clear that the boundary case counts are non-trivial and additional features that are more discriminative may be needed to improve the performance.

The numerical distribution of the errors across different databases (subsets of the dataset originating from different labs) is shown in Table 6. The databases are arranged in the increasing order of the sample size with database-c having

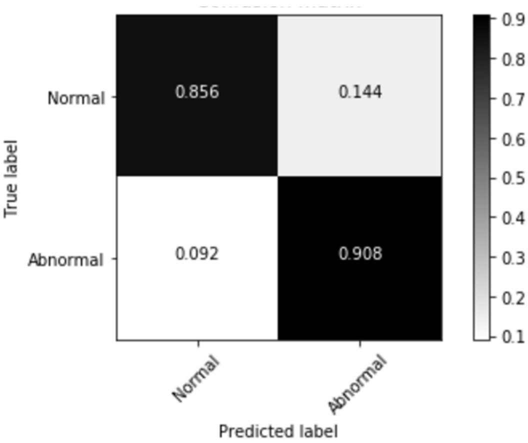

Fig. 7: Confusion matrix on one of the sample runs of our model

Table 5: Percentage of the test errors that are similar to the true classes

	Closer to negative samples	Closer to positive samples
False Positives	38.8%	61.1%
False Negatives	83.33%	16.66%

the least number of samples and database-e having the highest sample size. From Table 6, we can observe that the test error decreases as the samples size increases with the exceptions of database-b and database-f. The percentage error shows that except for database-e, all the other databases perform poorly in classifying the heart sounds. On examining the original distribution of heart sounds among different databases, the correlation between the percentage error and the sample size in each database is apparent. In the original dataset, database-e has the maximum number of heart sound recordings and it decreases with databases b, a, and f, to an extent that databases d and c have only 55 and 31 heart sound recordings respectively. Thus, we have many errors for databases which have fewer samples and few errors for database-e which has the highest number of heart sound recordings. As mentioned earlier, these databases are obtained from different healthcare facilities in which the recording instruments and locations of recording are different. Since the pattern of error in the instruments and the surrounding environment might be different for different healthcare centers, a model trained on only one particular database is more likely to perform poorly on the other. To build a more generalized model that performs well with data from different sources, the model should be trained on larger datasets from each of these sources. This would help capture the variations present in the data from different sources and should generalize well on a variety of heart sounds.

Table 6: Test error distribution among different subsets

Database Name	% Data distribution	% Test error
Database-c	0.95	33.33
Database-d	1.69	40.00
Database-f	3.51	52.94
Database-a	12.6	23.68
Database-b	15.12	44.44
Database-e	66.08	0.5

6 Comparison: Random Forest vs Other Classifiers

Apart from the random forest classifier, we have also explored two other classification algorithms: SVMs and logistic regression. The experimental settings were using the configuration 4 from Section 3.4 in terms of the majority class under sampling. Hyper parameters were fine-tuned using grid search. For this particular task, we found that SVMs are biased towards positive/abnormal class and more instances are predicted as abnormal thus resulting in better recall and lower precision. The loss in precision is nearly proportion to the gain in recall. As such, further exploration might be warranted in the future. Logistic regression also suffers from the same issue as with SVMs but the situation is much worse in terms of loss in precision. The results of these classifiers are shown in Table 7.

Table 7: Comparison: Random Forest, SVM and Logistic Regression

	Precision	Recall	F-score	Accuracy
Random Forest Classifier	0.637	0.912	0.749	0.887
Support Vector Machines	0.581	0.959	0.722	0.889
Logistic Regression	0.462	0.965	0.624	0.836

7 Limitations

Although our effort shed light on the precision-recall trade-off aspects in heart sound classification, we have the following limitations.

- We still do not have public access to the hidden test set that was actually used for evaluation during the 2016 PhysioNet/CinC challenge. Hence a direct comparison of our results against challenge participants is not possible.

The metric used is also different based on weights given to noisy examples. However, our accuracy of 88.7 in Table 4 is on par with other researchers' [16, 17] cross-validation experiments[4] on the public training data. Furthermore, our parameter tuning was focused on the objective of maximizing F-score (not accuracy) suitable for situations with class imbalance with minority positive class.
- Our model requires that the heart recording be long enough to have at least 2–3 heart cycles in it as the model generalizes well with more number of heart cycles, improving the accuracy of the system.
- Since there are various types of cardiovascular diseases, it is quite possible that the training samples are not representative of all the cardiac diseases.

8 Related Work

Here we outline prior efforts from the 2016 PhysioNet/CinC challenge participants. Potes et al. [11] employed the aggregate features we used in Section 3.3 to train a AdaBoost-abstain classifier composed of several weak learners, one for each feature. They also used four convolutional neural networks (CNNs) on each heart cycle, one for each of frequency ranges 25-45 Hz, 45-80 Hz, 80-200 Hz, and 200-400 Hz. The output of these four CNNs is flattened and input to a multi-layer perceptron. The final decision is made using a combination of the AdaBoost and CNN models. They have achieved recall of 94.24% and specificity of 77.81%. Rubin et al. [12] used the spectral features such as MFCC to obtain a two-dimensional time-frequency heat map representation. This 2-D heat map is used in training a deep convolutional neural network. With this approach they have achieved a high specificity of 93.1% and a low recall rate of 76.5%. Zabihi et al. [17] avoid the heart sound segmentation phase by using an ensemble of 20 feed forward neural networks to predict the final result by a voting mechanism. They used features based on the properties of the sound waves, extracted from time domain, frequency domain, and time-frequency domain signals to transform the input signal to a more meaningful representation before feeding it to the neural network. Although they avoided the segmentation process, they obtained comparable results with a specificity of 84.9% and recall of 86.9%.

9 Conclusion

In this paper, we present the details of supervised heart sound classification experiments we conducted using the 2016 PhysioNet/CinC challenge. Using random forests, SVMs, and logistic regression, we showed that a recall over 90% can be achieved and specifically using bagged random forests with under sampling we show that this can be done with a precision of 64%. Most of the features we used are inspired by the efforts in the signal processing community. However, based on error analysis experiments, we conclude that a richer feature space might

[4] Even this may not be exact comparison because the numbers of folds were different.

be needed to build better models especially in terms of increasing precision. As a next step, we could explore more complex ensembles using a wide variety of classification algorithms (including deep neural networks) to improve precision. Another area to be explored is to find the right combination of signal processing techniques that projects the input signal to a different feature space where the patterns are more clearly distinguishable. With more people working in this field and better performing systems, real time monitoring of the heart health could enable early detection of cardiovascular disease in low resource settings and decrease the mortality due to this disease.

Acknowledgements

We thank anonymous reviewers for their honest and constructive criticism of our paper. Our work is primarily supported by the National Library of Medicine through grant R21LM012274. We are also supported by the National Center for Advancing Translational Sciences through grant UL1TR001998. The content is solely the responsibility of the authors and does not necessarily represent the official views of the NIH.

References

1. American Heart Association. Heart disease and stroke statistics 2017. at-a-glance. `https://www.heart.org/idc/groups/ahamah-public/@wcm/@sop/@smd/documents/downloadable/ucm_491265.pdf`.
2. K. J. Archer and R. V. Kimes. Empirical characterization of random forest variable importance measures. *Computational Statistics & Data Analysis*, 52(4):2249–2260, 2008.
3. Bruno A. Olshausen. Aliasing. `http://redwood.berkeley.edu/bruno/npb261/aliasing.pdf`.
4. Cleveland Clinic. Heart and blood vessels: How does the heart beat. `https://my.clevelandclinic.org/health/articles/heart-blood-vessels-heart-beat`.
5. G. D. Clifford, C. Liu, B. Moody, D. Springer, I. Silva, Q. Li, and R. G. Mark. Classification of normal/abnormal heart sound recordings: the physionet/computing in cardiology challenge 2016. In *Computing in Cardiology Conference (CinC), 2016*, pages 609–612. IEEE, 2016.
6. 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, and H. E. Stanley. Physiobank, physiotoolkit, and physionet. *Circulation*, 101(23):e215–e220, 2000.
7. M. R. Hasan, M. Jamil, M. G. Rabbani, and M. S. Rahman. Speaker identification using mel frequency cepstral coefficients. In *3rd International Conference on Electrical and Computer Engineering*, pages 565–568, 2004.
8. C. Liu, D. Springer, Q. Li, B. Moody, R. A. Juan, F. J. Chorro, F. Castells, J. M. Roig, I. Silva, A. E. Johnson, et al. An open access database for the evaluation of heart sound algorithms. *Physiological Measurement*, 37(12):2181, 2016.
9. A. V. Oppenheim and G. C. Verghese. *Signals, systems and inference.* Pearson, 2015.

10. B. Ozenne, F. Subtil, and D. Maucort-Boulch. The precision–recall curve overcame the optimism of the receiver operating characteristic curve in rare diseases. *Journal of clinical epidemiology*, 68(8):855–859, 2015.
11. C. Potes, S. Parvaneh, A. Rahman, and B. Conroy. Ensemble of feature-based and deep learning-based classifiers for detection of abnormal heart sounds. In *Computing in Cardiology Conference (CinC), 2016*, pages 621–624. IEEE, 2016.
12. J. Rubin, R. Abreu, A. Ganguli, S. Nelaturi, I. Matei, and K. Sricharan. Classifying heart sound recordings using deep convolutional neural networks and mel-frequency cepstral coefficients. In *Computing in Cardiology Conference (CinC), 2016*, pages 813–816. IEEE, 2016.
13. T. Saito and M. Rehmsmeier. The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets. *PloS one*, 10(3):e0118432, 2015.
14. D. B. Springer, L. Tarassenko, and G. D. Clifford. Logistic regression-hsmm-based heart sound segmentation. *IEEE Transactions on Biomedical Engineering*, 63(4):822–832, 2016.
15. B. C. Wallace, K. Small, C. E. Brodley, and T. A. Trikalinos. Class imbalance, redux. In *Data Mining (ICDM), 2011 IEEE 11th International Conference on*, pages 754–763. IEEE, 2011.
16. B. M. Whitaker, P. B. Suresha, C. Liu, G. Clifford, and D. Anderson. Combining sparse coding and time-domain features for heart sound classification. *Physiological Measurement*, 2017.
17. M. Zabihi, A. B. Rad, S. Kiranyaz, M. Gabbouj, and A. K. Katsaggelos. Heart sound anomaly and quality detection using ensemble of neural networks without segmentation. In *Computing in Cardiology Conference (CinC), 2016*, pages 613–616. IEEE, 2016.