Medical & Biological Engineering & Computing manuscript No.
(will be inserted by the editor)

Noninvasive Diagnosis of Pulmonary Hypertension Using Heart Sound Analysis

Aaron Dennis · Andrew D. Michaels · Patti Arand · Dan Ventura

Received: date / Accepted: date

Abstract Right-heart catheterization is the most accurate method for measuring pulmonary artery pressure (PAP). It is an expensive, invasive procedure, exposes patients to the risk of infection, and is not suited for long-term monitoring situations. Medical researchers have shown that PAP influences the characteristics of heart sounds. This suggests that heart sound analysis is a potential method for the noninvasive diagnosis of pulmonary hypertension. We describe the development of a prototype system, called PHD (pulmonary hypertension diagnoser), that implements this method. PHD uses patient data with machine learning algorithms to build models of how pulmonary hypertension affects heart sounds. Data from 20 patients was used to build the models and data from another 31 patients was used as a validation set. PHD diagnosed pulmonary hypertension in the validation set with 77% accuracy and 0.78 area under the receiver-operating-characteristic curve.

Keywords Learning systems · Blood pressure · Cardiovascular system

1 Introduction

Measuring the pulmonary artery pressure (PAP) is important because it is “a very useful parameter for the clinical evaluation of many cardiac diseases” [1]. Right-heart catheterization and Doppler echocardiography can both be used to measure PAP. These methods are not ideal for some patients and under some circumstances; consequently, heart sound analysis is being studied as an alternative method for PAP estimation. In this paper we develop a system that diagnoses abnormally high PAP (*i.e.*, pulmonary hypertension, or PH) using heart sound analysis.

Right-heart catheterization gives the most reliable and accurate measurement of PAP [1][11]. It is performed by threading a Swan-Ganz catheter through a vein until it reaches the pulmonary artery, at which point a PAP measurement can be made. Disadvantages of this approach include high expense, risk of infection, and risk of physical harm to internal bodily structures.

Doppler echocardiography uses ultrasound technology and the Doppler effect to measure the speed and direction of blood flow within the heart. Doppler echocardiography is noninvasive, safe, and relatively cheap. The disadvantage is that it cannot be used to estimate PAP “in approximately 50% of patients with normal PAP, 10-20% of patients with increased PAP, and 34-76% of patients with chronic obstructive pulmonary disease” [15].

Heart sound analysis is noninvasive, inexpensive, safe, can be used on most if not all patients, and may be automated using computer software. It has the potential to provide PAP estimates without the disadvantages of right-heart catheterization and Doppler echocardiography. However, heart sound analysis is still in an experimental stage and has not yet matured enough to replace the other methods of PAP estimation.

Heart sound analysis is based on the idea that PAP levels have an effect on the characteristics of the heart sounds, especially S2 (the second heart sound). Theo-

A. Dennis · D. Ventura
Department of Computer Science, Brigham Young University, Provo, UT 84602, USA
E-mail: adennis@byu.edu

A. D. Michaels
Department of Internal Medicine, University of Utah, Salt Lake City, UT 84132, USA

P. Arand
Inovise Medical, Inc., Beaverton, OR 97008, USA

retical considerations and experimental results support this idea [1][4][15]. For example, Aggio notes that PAP is “known to influence the characteristics of the second heart sound (S2): a rise of PAP is associated with an enhancement of its pulmonary component” [1]. This and other relationships can be exploited to estimate PAP by analyzing heart sounds.

The system described in this paper uses machine learning algorithms to implement heart sound analysis for pulmonary hypertension diagnosis. The relationship between PAP and heart sounds is not fully understood and may be quite complex; consequently, an analytical solution would be difficult to implement. In contrast, machine learning algorithms can infer from example patient data complex models of the PAP/heart sound relationship.

1.1 Previous Work

Several studies have measured patient data (*e.g.*, S2 features paired with PAP values) and then modeled this data using curve fitting techniques. Statistical measures such as the correlation coefficient are used to measure how well the curve models the data. A primary goal in these studies is to determine which heart sound features can be used to build good models of the data. Aggio *et al.* studied various characteristics of the frequency spectrum of P2 (the pulmonic component of S2) [1]. Chen *et al.* looked at additional S2 frequency spectrum characteristics [4]. Xu *et al.* looked at the splitting interval between A2 (the aortic component of S2) and P2 [15]. Many of the features from these studies will be used in this paper (see Sect. 2.2 for specifics).

The approach taken here is different from the typical approach used in previous studies. We use machine learning algorithms to infer models of the data instead of curve fitting. We use the models to classify test data instead of measuring the correlation between a model and the data. Classifying test data provides us with an estimate of the model's predictive accuracy on future patients.

Tranulis *et al.* used a time-frequency representation of S2 to train a multilayer perceptron for PAP estimation and patient diagnosis [12]. This approach is similar to our approach, building a data model using a machine learning algorithm and estimating the model's predictive accuracy using test data. Unfortunately, their model's reported accuracy is overly optimistic; it does not reflect the model's performance on real-world patients. This is because the heart sounds of each patient in the study were randomly shuffled and then split into two groups, with one group ending up in the training set and the other in the test set. In a real-world situation, a model will not have been trained using data from a patient that needs to be diagnosed. In this paper we will use two disjoint sets of patients to create training and test sets.

1.2 Patient Data

Patient data collection was done under IRB approval and all patients provided written informed consent. An Audicor machine (Inovise Medical, Inc.) was used to record phonocardiogram (PCG) and electrocardiogram (ECG) traces from 51 patients undergoing right-heart catheterization. Patient PAP values were measured using a Swan-Ganz catheter, and a patient is considered as having pulmonary hypertension if his/her mean PAP is greater than or equal to 25 mmHg. The 51 patients are split into two sets: a set of 20 patients and another set of 31 patients. The set of 31 patients is used as a hold-out set for final validation of our models. The mean and standard deviation ($M \pm SD$) of the mean PAP values for the 20 patients is (23.8 ± 10.9); for the 31 patients it is (25.2 ± 11.1); and for all 51 patients it is (24.6 ± 10.9).

2 Method

This paper describes the development of a prototype pulmonary hypertension diagnosis system which we call PHD (*p*ulmonary *h*ypertension *d*iagnoser). We constrain the design space of PHD to three sets of parameters: chest wall location, heart sound features subset, and machine learning algorithm. Designing PHD is equivalent to choosing values for these parameters. The next three subsections describe the parameters and the values that can be assigned to them.

Choosing values for the three design parameters is complicated by the fact that the parameters are not independent of one another. We cannot decompose the design task into a search for the optimal value of the first parameter, then the second, then the third. We are forced to evaluate many parameter settings to find a good one.

2.1 Chest Wall Locations

Heart sound characteristics change depending on the chest wall location that is used to record the sound. When building and testing classifier models for PHD we use data from a single chest wall location, which allows us to compare the performance of PHD as a function of chest wall location (see Fig. 2) and determine which locations are most useful for PH diagnosis. The PCG and ECG traces were recorded for each of the following five chest wall locations: the V3 position, the V4 position, the second interspace left parasternal position, the left parasternal pulmonic region, and the right parasternal aortic region.

2.2 Heart Sound Features

We extract 46 features for each recorded heart sound in the PCG traces. These candidate features include many that are described in the medical literature, some that are derivatives of these features, features calculated by Inovise, and some miscellaneous features. A summary description of all the candidate heart sound features appears in Table 1. PHD's performance is dependent on the features used; finding a suitable subset of these features is an important design challenge.

Engineers at Inovise provided us with proprietary heart sound features which we have named as follows: c_{S1}, i_{S1}, w_{S1}, c_{S2}, i_{S2}, w_{S2}, i_{S3}, s_{S3}, i_{S4}, and s_{S4}. These features correspond to the intensity or width of the four heart sounds (S1-S4). S3 and S4 are, respectively, the third and fourth heart sound. S4 is always abnormal and S3 is an early sign of heart disease in patients over 40 years in age.

We calculate the seven spectral features described by Chen *et al.* [4] (two of which were also studied by Aggio *et al.* [1]). These include the dominant frequencies of S2, A2, and P2 (F_{S2}, F_{A2}, and F_{P2} respectively), the quality of resonance of A2 and P2 (Q_{A2} and Q_{P2} respectively), and the following ratios: F_{P2}/F_{A2} and Q_{P2}/Q_{A2}. Mathematical descriptions of these features appear in Table 1.

The splitting interval of the second heart sound and ventricular systole durations are also extracted. The splitting interval (SI) and normalized splitting interval (NSI) were studied in [15]. The SI is the time between the beginning of A2 and beginning of P2. The NSI is the SI normalized by the heart rate.

Left and right ventricle systole durations are estimated and used as features. These features are selected based on the idea that a higher PAP leads to a prolonged systole duration and/or a greater percent of the cardiac cycle being required for systole. The hypothesis is that a longer period of time is required to pump blood through high-pressure, stiff arteries and capillaries.

The ventricle systole durations are calculated as follows. Either the R-wave time or the S1 start time is used to mark the start of systole. The end of left and right ventricle systole are marked by the start of the A2 sound and the start of the P2 sound, respectively. This results in the following features, where the systole begin time is indicated by the subscript and the systole end time is indicated by the superscript: D_R^{A2}, D_R^{P2}, D_{S1}^{A2}, D_{S1}^{P2}. The percent of the heartbeat duration taken by systole is calculated by dividing the systole duration features by the length of the corresponding heartbeat (δ_{HB}). These features are denoted with a tilde sign, and include the following: $\tilde{D}_R^{A2}$, $\tilde{D}_R^{P2}$, $\tilde{D}_{S1}^{A2}$, $\tilde{D}_{S1}^{P2}$.

Table 1 Heart Sound Features. PHD uses these features to diagnose PH. In this table *sig* can be one of the following heart sound signals: HB, S1, S2, A2, or P2, where HB is the whole heart sound signal. Terms such as $t_{P2start}$ are the onset times of the indicated heart sound component. t_R is the ECG R-wave time and δ_{RR} is the time between two successive R-waves.

Category	Features	Description
Inovise Features	c_{S1}, i_{S1}, w_{S1}, c_{S2}, i_{S2}, w_{S2}, i_{S3}, s_{S3}, i_{S4}, s_{S4}	Heart Sound Intensity/Width
Dominant Frequency[a]	F_{HB}, F_{S1}, F_{S2}, F_{A2}, F_{P2}	$\underset{k}{\text{argmax}}\ \mathcal{F}(sig)_k$
Quality of Resonance[b]	Q_{HB}, Q_{S1}, Q_{S2}, Q_{A2}, Q_{P2}	$F_{sig}/(R_{sig} - L_{sig})$
Power[c]	P_{HB}, P_{S1}, P_{S2} P_{A2}, P_{P2}	$\frac{1}{T}\sum\limits_{x \in sig} \lvert x \rvert^2$
Splitting Interval	SI_{S1}	$t_{T1start} - t_{M1start}$
	SI_{S2}	$t_{P2start} - t_{A2start}$
	NSI_{S1}	$\frac{SI_{S1} \times HR}{600}$
	NSI_{S2}	$\frac{SI_{S2} \times HR}{600}$
Ratios	$R_{F_{A2}}^{F_{P2}}$	F_{P2}/F_{A2}
	$R_{Q_{A2}}^{Q_{P2}}$	Q_{P2}/Q_{A2}
	$R_{P_{A2}}^{P_{P2}}$	P_{P2}/P_{A2}
	$R_{P_{S2}}^{P_{A2}}$	P_{A2}/P_{S2}
	$R_{P_{S2}}^{P_{P2}}$	P_{P2}/P_{S2}
	$R_{P_{S1}}^{P_{A2}}$	P_{A2}/P_{S1}
	$R_{P_{S1}}^{P_{P2}}$	P_{P2}/P_{S1}
	$R_{P_{S1}}^{P_{S2}}$	P_{S2}/P_{S1}
Systole Duration	D_R^{A2}	$t_{A2start} - t_R$
	D_R^{P2}	$t_{P2start} - t_R$
	D_{S1}^{A2}	$t_{A2start} - t_{S1start}$
	D_{S1}^{P2}	$t_{P2start} - t_{S1start}$
	$\tilde{D}_R^{A2}$	D_R^{A2}/δ_{RR}
	$\tilde{D}_R^{P2}$	D_R^{P2}/δ_{RR}
	$\tilde{D}_{S1}^{A2}$	D_{S1}^{A2}/δ_{RR}
	$\tilde{D}_{S1}^{P2}$	D_{S1}^{P2}/δ_{RR}
Heart Rate[d]	HR	$m/\sum_{i=1}^{m} \delta_{RR}^{i}$

[a] $\mathcal{F}(sig)_k$ is the k^{th} frequency sample of the DFT of *sig*.

[b] R_{sig} and L_{sig} are, respectively, the frequencies to the right and to the left of F_{sig} at which the value of the DFT drops to half of the maximum.

[c] T is the length of *sig*.

[d] m is the number of surrounding heartbeats to include in the calculation.

We also extract general audio features of the heart sounds. Specifically, the power of the S2, A2, and P2 sounds (P_{S2}, P_{A2}, and P_{P2} respectively) are calculated.

The following ratios are also calculated: P_{P2}/P_{A2}, P_{A2}/P_{S2}, P_{P2}/P_{S2}, P_{A2}/P_{S1}, P_{P2}/P_{S1}, and P_{S2}/P_{S1}.

We calculate additional features mainly because it is simple to do so. The dominant frequency, Q-factor, and power for the whole heart sound and for the S1 sound is added (F_{HB}, Q_{HB}, P_{HB}, F_{S1}, Q_{S1}, and P_{S1}). The splitting interval and normalized splitting interval of the S1 sound (SI_{S1} and NSI_{S1}) is added as well as the quality of resonance of the S2 sound and the heart rate (Q_{S2} and HR).

2.3 Learning Algorithms

We select PHD's classification model generator from a set of five candidate machine learning algorithms: decision tree, k–nearest neigbhors, multilayer perceptron, naive Bayes, and support vector machine. These are all well-known and well-understood algorithms that perform inference and build models in different ways.

The decision tree algorithm (J48) is an implementation of the C4.5 algorithm developed by Quinlan [10]. It generates the decision tree by splitting on features that maximize information gain. The k–nearest neighbors (KNN) algorithm is an instance-based learning method [2] that builds a local model using lazy evaluation. The multilayer perceptron (MLP) is a connectionist model that is trained using the Backpropogation algorithm. Naive Bayes (NB) is a probabilistic model that makes the assumption of conditional independence between feature given the target value [8] [6]. Support vector machines (SMO) find a margin-maximizing decision surface (a hyperplane) within a high-dimensional space to which input vectors are (implicitly) mapped in a non-linear or linear fashion [9] [3]. In this paper we used a linear support vector machine, which can be thought of as an optimized linear perceptron. For all algorithms we used the implementation found in WEKA [13].

2.4 Experiments

In the experiments described in this section we build PHD systems and evaluate their performance. The training data for the validation experiments (Sect. 2.4.4) comes from 20 of the 51 patients in this study and the test data comes from the other 31 patients. In all other experiments the training and test data come from the set of 20 patients. Each PHD system is built using a selected location, a set of features, and a learner. We define the term "PHD configuration", or just "configuration" to mean a tuple that includes a location, list of features, and a learner. It may also refer to a PHD system built using the tuple.

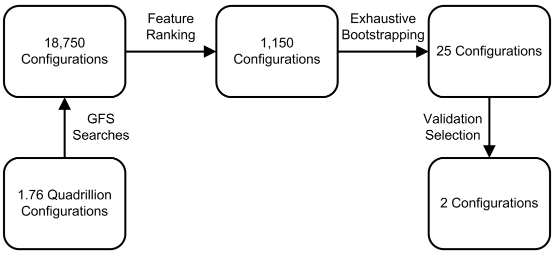

Fig. 1 Experiments Overview. We reduce the number of candidate PHD configurations using results from a series of experiments.

The design space for PHD is large; the total number of possible PHD configurations is about 1,759 trillion ($5\ locations \times 2^{46} - 1\ feature\ subsets \times 5\ learners$). We use several methods to reduce the size of this search space; Fig. 1 gives an overview of this process. We apply a feature selection algorithm (Sect. 2.4.1) to select a subset of the possible configurations. Results from the search are combined (Sect. 2.4.2) to form an even smaller subset. From this smaller subset we select 25 configurations using a modified form of bootstrapping (Sect. 2.4.3) that we call exhaustive bootstrapping. Two of these 25 configurations outperform the rest at classifying the 31 validation-set patients (Sect. 2.4.4).

The experiments in Sections 2.4.1, 2.4.3, and 2.4.4 estimate the real world performance of many PHD configurations. Cross-validation was used in Section 2.4.1 and exhaustive bootstrapping was used in Section 2.4.3. In Section 2.4.4 configurations were trained on a training set and tested on a validation set.

With an unlimited amount of (i.i.d.) data, the performance estimation method of Section 2.4.4 would suffice. However, this method produces biased results when used with a small dataset, which is the case in these experiments. We have data from 51 patients, 31 of which were reserved for the experiment in Section 2.4.4. Even if no patients were held out, 51 patients is still a small dataset size. To get less biased performance estimates using our small dataset, we use the (non-ideal) cross-validation and bootstrapping methods [5].

Cross-validation and bootstrapping differ in the way that the dataset is split in each iteration. Cross-validation "folds" the dataset into roughly equal-sized sets. One set is used as the test set while the other sets are combined to form the training set. This is repeated, with a different set being used as the test set, until all sets have been used as the test set exactly once. Bootstrapping creates a training set by randomly selecting a given percent of the data; it creates a test set using the rest of the data. This is repeated as often as necessary.

2.4.1 Greedy Forward-Selection Search

We use a feature subset selection algorithm [7] to select a group of feature subsets from the feature subset search space, whose size is $2^{46} - 1$. We use the wrapper approach [7] and 10-fold cross-validation for evaluating subsets. Using the wrapper approach makes sense since the performance of PHD is dependent on both the selected learning algorithm and the selected feature subset.

The search method for our feature subset selection algorithm is a greedy forward-selection (GFS) search. A forward-selection search begins with an empty set of features; features are added to this set as the search progresses. This differs from the backward-elimination search which starts with all possible features and eliminates features from the set during the search. The search is greedy because it iteratively adds the single feature that will improve performance the most; it does not consider adding higher-order sets of features.

We perform 25 GFS searches, one for each of the location/learner pairs. The searches run for 30 iterations, producing 30 feature subsets ($25 \times 30 = 750$ subsets in all).

2.4.2 Feature Ranking

We rank all 46 features from Section 2.2, producing a ranked feature list, by assigning a score to each feature. This is done using the 25 feature lists that result from the GFS searches. The feature scoring uses two assumptions about features in the 25 feature lists. The first assumption is that features chosen early in a GFS search are more important than features chosen later in the search. The second assumption is that features appearing in more of the 25 lists are more important than features appearing in fewer of the 25 lists. (Each of the 25 lists contained 30 of the 46 features, so every feature did not appear in every list.)

The following equation is used to score a feature:

$$f_{score} = \sum_{l \in L} 30 - f^{l}_{rank}$$

where f_{score} is the feature score, L is the set of lists in which the feature appears, and f^{l}_{rank} is the position in the list, l, in which the feature appears.

2.4.3 Exhaustive Bootstrapping

The configurations that we evaluate in the exhaustive bootstrapping experiments use one of only 46 candidate feature subsets (a large reduction from the original $2^{46} - 1$ candidate subsets). The subsets are created using the ranked feature list. The first feature in the list is the first subset; the first two features in the list are the second subset, and so on with the 46th subset being all 46 features. Combined with the five locations and five learners, we end up still having $46 \times 5 \times 5 = 1150$ candidate configurations to evaluate.

We use a modified version of bootstrapping (which we call exhaustive bootstrapping) to evaluate the configurations. We repeatedly split the set of 20 patients into training and test sets. Each split consists of 18 patients in the training set and two patients in the test set. Unlike normal bootstrapping where splits are formed randomly and repeatedly a given number of times, we systematically split the data in all possible ways in which there are two patients in the test set. Splitting patients in this way leads to $\binom{20}{2} = 190$ different splits, creating 190 different test sets each of size two. Each patient appears in 19 of the test sets and is paired with a different patient each time. Consequently we calculate 19 performance evaluations per patient and each evaluation is associated with a slightly different training set.

Exhaustive bootstrapping is used to mitigate the effects of working with a small dataset. It allows us to produce more configuration performance estimates than cross-validation or a typical use of bootstrapping. More estimates may lead to a more accurate final estimate (which is an average of the estimates). Also, unlike regular bootstrapping, exhaustive bootstrapping avoids producing duplicate evaluation results by preventing two results from being produced by the same training set and test set.

The exhaustive bootstrapping experiments produce 380 performance evaluations for each of the 1150 configurations. The average of each set of 380 evaluation results is taken as the classification accuracy for its associated configuration. Also, the 380 accuracy measures are used to form a receiver-operating-characteristic (ROC) curve and to calculate the area under the curve (AUC). For each location/learner pair, we select the configuration with the highest AUC.

2.4.4 Validation

We classify the holdout set of 31 patients using the 25 selected configurations. This is done by having each configuration classify all heartbeats from each patient; a patient is classified as sick (i. e. having pulmonary hypertension) if the percentage of the patient's heartbeats that are classified as sick crosses a calculated threshold.

Each of the 25 configurations uses a different threshold, which is calculated using the configuration's associated ROC curve. Each point on the curve is associated with a false-positive rate (FPR), a true-positive rate

(TPR), and a threshold. We want a low FPR and a high TPR so we calculate a score for each point using the following equation: $score = TPR - FPR$. The configuration uses the threshold with the highest score.

3 Results

3.1 Ranked Feature List

The feature lists resulting from the GFS searches were combined (see Sect. 2.4.2) to create a ranked feature list, shown in Table 2. The features are ranked in descending order from most important to least important feature.

Physiological considerations lead us to expect that features extracted from the S2 sound are helpful in classifying a patient. On the other hand we expect that other features, such as those extracted from the S1 sound or the whole heart sound, are less helpful. The ranked feature list confirms these expectations, although some exceptions do occur.

The features extracted from the whole heart sound were not very helpful. F_{HB}, Q_{HB}, P_{HB}, and HR are all ranked in the second half of the list. The features wholly-dependent on the S1 sound (i_{S1}, c_{S1}, Q_{S1}, P_{S1}, SI_{S1}, w_{S1}, F_{S1}) were also all ranked in the second half of the list. Together, these features make up 11 of the 23 features in the second-half of the list.

The ventricular systole duration estimates (D_{S1}^{A2}, D_{S1}^{P2}, D_{R}^{A2}, and D_{R}^{P2}) were not predictive, as indicated by their ranking in the ranked feature list. They occupied the 41^{st} ranking and the last three rankings. However, simply dividing these duration times by the whole heartbeat time increased the predictivity. The features $\tilde{D}_{R}^{A2}$, $\tilde{D}_{R}^{P2}$, $\tilde{D}_{S1}^{A2}$, and $\tilde{D}_{S1}^{P2}$ occupy the 5^{th}, 7^{th}, 16^{th}, and 25^{th} rankings. Thus, the percentage of the heartbeat taken for ventricular systole was much more predictive than the absolute duration time of ventricular systole.

Statistical analysis in [4] found that, for their data, F_{A2}, Q_{A2}, and Q_{P2}/Q_{A2} did not have a significant influence on pulmonary artery systolic pressure. The ranked feature list gives these features more credit, ranking them, respectively, at 13^{th}, 8^{th}, and 6^{th}.

Many of the features in the ranked feature list are ranked in accordance with reasonable expectations based on medical knowledge. The $R_{P_{A2}}^{P_{P2}}$ feature is a powerful predictive bedside tool for diagnosing pulmonary hypertension. SI_{S2} is also expected to be a useful feature. On the other hand, heart rate, systolic ejection period, and S1 splitting are not expected to be useful and this is reflected in the ranked feature list. One surprise is that the heart sound resonance (the Q-features) did not seem to have more predictive value. Q_{A2} was the only Q-feature with good predictive value.

Table 2 Ranked Feature List. This list of ranked features was produced by combining the feature lists from the 25 GFS searches.

Rank	1	2	3	4	5	6
Feature	$R_{P_{A2}}^{P_{P2}}$	s_{S4}	$R_{F_{A2}}^{F_{P2}}$	i_{S4}	$\tilde{D}_{R}^{A2}$	$R_{Q_{A2}}^{Q_{P2}}$
Rank	7	8	9	10	11	12
Feature	$\tilde{D}_{R}^{P2}$	Q_{A2}	$R_{P_{S2}}^{P_{P2}}$	SI_{S2}	s_{S3}	$R_{P_{S2}}^{P_{A2}}$
Rank	13	14	15	16	17	18
Feature	F_{A2}	c_{S2}	w_{S2}	$\tilde{D}_{S1}^{A2}$	P_{S2}	P_{A2}
Rank	19	20	21	22	23	24
Feature	F_{P2}	$R_{P_{S1}}^{P_{P2}}$	Q_{P2}	$R_{P_{S1}}^{P_{A2}}$	P_{P2}	i_{S3}
Rank	25	26	27	28	29	30
Feature	$\tilde{D}_{S1}^{P2}$	P_{HB}	$R_{P_{S1}}^{P_{S2}}$	i_{S1}	Q_{S2}	c_{S1}
Rank	31	32	33	34	35	36
Feature	NSI_{S2}	Q_{S1}	P_{S1}	Q_{HB}	F_{S2}	SI_{S1}
Rank	37	38	39	40	41	42
Feature	i_{S2}	w_{S1}	F_{HB}	HR	D_{S1}^{A2}	F_{S1}
Rank	43	44	45	46		
Feature	NSI_{S1}	D_{S1}^{P2}	D_{R}^{A2}	D_{R}^{P2}		

3.2 Parameter Performance

In this subsection we average the results from the exhaustive bootstrapping experiments across each parameter of the system design space. Doing this allows us to compare the possible parameter values to determine which values, in general, lead to good performance and which do not.

Fig. 2 plots the average accuracy and average AUC for all configuration evaluations as a function of location. This graph clearly indicates that Location 3 and Location 4 are not as suited for PH diagnosis as the other three locations. Location 1 and Location 5 have roughly equivalent performance and Location 2 is only slightly worse.

Fig. 3 plots the average accuracy and average AUC for all configuration evaluations as a function of learner. The graph indicates that the J48 learning algorithm performs at a much lower level than the other learners. Naive Bayes is clearly the best-performing learner. KNN and MLP perform at about the same level. SMO's accuracy is comparable to KNN and MLP but its average AUC measure is significantly worse.

Fig. 4 plots the average accuracy and average AUC for all configuration evaluations as a function of feature subset. As the size of the feature subset grows, the performance initially increases quickly and then gradually decreases.

It is interesting to note which features, when added to the feature subset, cause a significant increase in performance. The first major increase happens when s_{S4} is

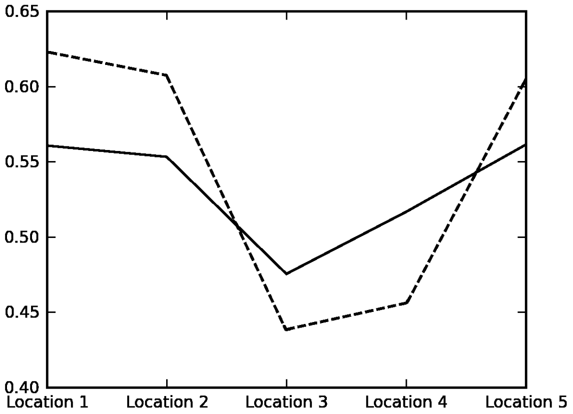

Fig. 2 Location Performance. Results from the experiments described in Section 2.4.3 are grouped based on location. The average accuracy (solid line) and average AUC values (dashed line) are plotted here.

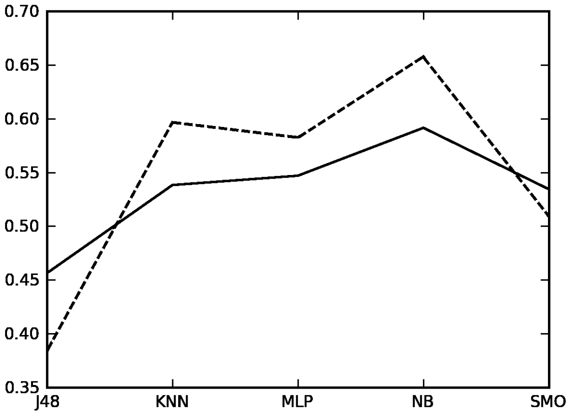

Fig. 3 Learner Performance. Results from the experiments described in Section 2.4.3 are grouped based on learner. The average accuracy (solid line) and average AUC values (dashed line) are plotted here.

added. The second increase is due to adding i_{S4}. This is somewhat surprising since most of our attention is on the S2 features, but the presence of an S3 and/or S4 sound can be useful in assessing pulmonary hypertension.

The third major increase in performance occurrs when SI_{S2} is added to the feature subset. We expect this feature to be useful in PH diagnosis because of its demonstrated utility elsewhere [15]. The fact that it initiated a large a jump in performance is further evidence of its usefulness in PH diagnosis.

Three more minor increases in performance are initiated by the addition of c_{S2}, w_{S2}, and P_{S2}. Each of these is a feature of the S2 heart sound alone. This supports the hypothesis that features of the S2 heart sound are useful in PH diagnosis.

Table 3 Validation Results: Accuracy and AUC. We classified 31 patients (the holdout set) using each of the 25 selected configurations. The accuracy (number on the left) and AUC (number on the right) are shown for each configuration. The AUC values have been multiplied by 100 to make them easier to read.

	J48	**KNN**	**MLP**	**NB**	**SMO**
Location 1	55/55	58/52	65/66	**77/78**	65/66
Location 2	52/58	61/67	58/61	55/59	58/62
Location 3	42/50	42/50	48/53	45/51	61/66
Location 4	55/47	39/45	58/61	68/67	65/59
Location 5	55/55	42/50	65/63	65/62	**74/74**

3.3 Holdout Set Classification

Table 3 shows results from the validation experiments (see Sect. 2.4.4). In these experiments we classify the 31 holdout patients using the 25 selected configurations. The accuracy and AUC are shown for each configuration. These numbers estimate the real-world performance of each configuration and we analyze them to select the best-performing configurations. The name L1NB is used to refer to the configuration associated with location 1 and the NB algorithm. Similar names are used for other configurations.

It is obvious from looking at Table 3 that L1NB is the best configuration and that L5SMO is only a little worse. The selection of L1NB and L5SMO is somewhat surprising because they differ from the results of the exhaustive bootstrapping experiments, which predict a high-performing L5NB and an average-performing L5SMO. The validation experiment results show L5NB doing worse than expected and L5SMO doing better than expected.

One possible reason for the success of L1NB and L5SMO is that both used learning algorithms that avoid overfitting the training data. Overfitting is a particular challenge when the training data is sparse, which is the case in our experiments. The naive Bayes algorithm and SMO algorithm avoid overfitting by producing simple decision surfaces (a linear SVM was used in L5SMO) that are unable to conform too closely to the training data. For example, the decision surface used by L5SMO is a hyperplane that cuts through the input feature space.

A PHD system built using L1NB, L5SMO, or one of the other configurations would, of course, need further testing and verification before enough confidence could be placed in the system to use it in real-world situations. This is another consequence of not having enough data in our datasets. However, our results give us confidence that a dependable PHD system can be built.

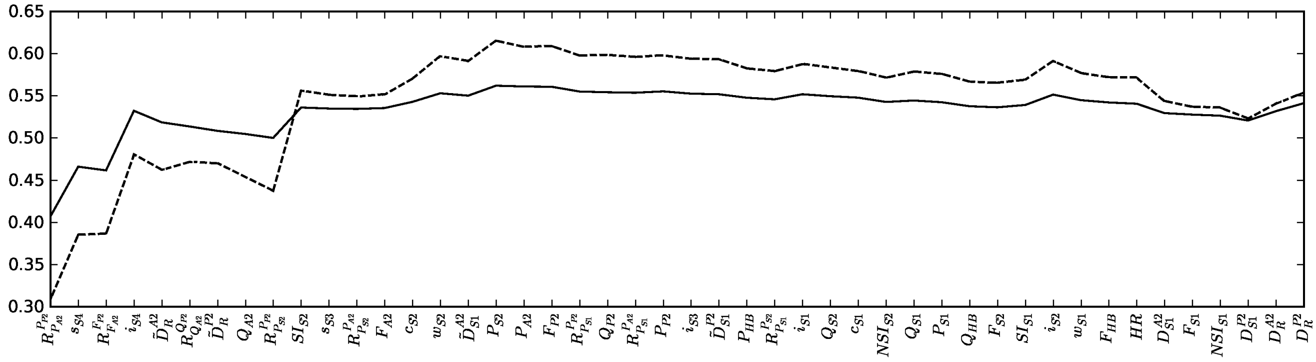

Fig. 4 Feature Performance. Results from the experiments described in Section 2.4.3 are grouped based on feature subset. The average accuracy (solid line) and average AUC values (dashed line) are plotted here.

4 Conclusion

Using tools from machine learning, we developed a prototype heart sound analysis system for noninvasive PH diagnosis, a system that could help lower diagnostic costs by replacing the use of right-heart catheterization on some patients. Developing PHD involved running several experiments geared toward answering three questions. What chest wall location should be used in recording heart sounds? What learning algorithm should be used to diagnose PH? What set of heart sound features should be used as input to the learning algorithm? The experiments resulted in at least two promising configurations of the PHD system.

We did not focus much effort on calculating high-quality feature values from the heart sounds. Typically the quickest and easiest method was used. In particular, our method of calculating SI_{S2} is not as advanced as the methods described in [14] and [15]. Incorporating these methods into PHD could lead to better SI_{S2} values and, presumably, better performance. This is true of the other heart sound features as well. We also made little effort to throw away features that could be considered noisy. We did not hand-pick the heartbeats we used in the datasets, instead trusting the machine learning algorithms to perform well despite the noisy data; but noise removal may improve the performance of a future PHD system.

The point is that we spent little time and effort on performing preprocessing of any kind. The fact that PHD performs as well as it does without any special tweaking speaks to its robustness. Moreover, various preprocessing steps may improve PHD's performance and reliability.

The promising configurations found in this paper all use somewhat large feature subsets (20 to 30 features). Finding a smaller feature subset that still produces good performance is an important goal for future work. Doing so would aid in the development of a physiological theory for how and why PHD works and would increase PHD's interpretability.

Currently, PHD does not try to estimate a precise PAP value; it simply tries to determine whether the PAP is above or below a certain threshold, where the threshold is the pressure above which a patient is considered to have pulmonary hypertension. It would be useful for future systems to produce an actual PAP value instead. This would provide doctors with valuable information about the severity of a patient's pulmonary hypertension or about the likelihood of a patient developing pulmonary hypertension.

As a quick attempt at implementing a PHD system that produces PAP values, we trained an L5MLP configuration using PAP values instead of pulmonary hypertension diagnoses. We measured the performance of the L5MLP configuration using the standard error of estimate (SEE[1]). On the holdout set of 31 patients it has an SEE of 11.7 mmHg. On the training set (the set of 20 patients) it has an SEE of 5.1 mmHg, which is better than the results reported by Xu *et al.* [15]; the SEE for the humans in that study is 5.8 mmHg. These results indicate that one focus of future work should be the development of a reliable and accurate PHD system that produces PAP values.

The configurations in this paper need to undergo more testing before being used in a clinical setting. This will require more patient data. Having more training data to train the classifiers would potentially lead to better classifiers. And testing the configurations on larger test sets is needed for calculating more reliable and more accurate performance estimates.

[1] $SEE = \sqrt{\sum (PAP_{est} - PAP_{act})^2/(N-2)}$, where PAP_{est} is the estimated PAP value, PAP_{act} is the actual PAP value, and N is the number of estimate/actual PAP value pairs.

References

1. Aggio S, Baracca E, Longhini C, Brunazzi C, Longhini L, Musacci G, Fersini C (1990) Noninvasive estimation of the pulmonary systolic pressure from the spectral analysis of the second heart sound. Acta Cardiologica 45(3):199–202, PMID: 2368539
2. Aha DW, Kibler D, Albert MK (1991) Instance-based learning algorithms. Machine Learning 6(1):37–66, DOI 10.1007/BF00153759
3. Burges CJC (1998) A tutorial on support vector machines for pattern recognition. Data Mining and Knowledge Discovery 2(2):121–167, DOI http://dx.doi.org/10.1023/A:1009715923555
4. Chen D, Pibarot P, Honos G, Durand L (1996) Estimation of pulmonary artery pressure by spectral analysis of the second heart sound. The American Journal of Cardiology 78(7):785–789, DOI 10.1016/S0002-9149(96)00422-5
5. Isaksson A, Wallman M, Göransson H, Gustafsson MG (2008) Cross-validation and bootstrapping are unreliable in small sample classification. Pattern Recognition Letters 29(14):1960–1965, DOI http://dx.doi.org/10.1016/j.patrec.2008.06.018
6. John G, Langley P (1995) Estimating continuous distributions in Bayesian classifiers. In: Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence, Morgan Kaufmann, pp 338–345
7. Kohavi R, John GH (1997) Wrappers for feature subset selection. Artificial Intelligence 97(1-2):273–324, DOI http://dx.doi.org/10.1016/S0004-3702(97)00043-X
8. Mitchell TM (1997) Machine Learning. McGraw-Hill, New York
9. Platt JC (1999) Fast training of support vector machines using sequential minimal optimization. In: Advances in kernel methods: support vector learning, MIT Press, pp 185–208
10. Quinlan JR (1993) C4.5: programs for machine learning. Morgan Kaufmann Publishers Inc.
11. Staff MC (2008) Pulmonary hypertension. http://www.mayoclinic.com/health/pulmonary-hypertension/DS00430
12. Tranulis C, Durand L, Senhadji L, Pibarot P (2002) Estimation of pulmonary arterial pressure by a neural network analysis using features based on time-frequency representations of the second heart sound. Medical and Biological Engineering and Computing 40(2):205–212, DOI 10.1007/BF02348126
13. Witten IH, Frank E (2005) Data Mining: Practical machine learning tools and techniques, 2nd edn. Morgan Kaufmann, San Francisco
14. Xu J, Durand L, Pibarot P (2001) Extraction of the aortic and pulmonary components of the second heart sound using a nonlinear transient chirp signal model. IEEE Transactions on Biomedical Engineering 48(3):277–283, DOI 10.1109/10.914790
15. Xu J, Durand L, Pibarot P (2002) A new, simple, and accurate method for non-invasive estimation of pulmonary arterial pressure. Heart (British Cardiac Society) 88(1):76–80, PMID: 12067952