DOKUZ EYLÜL UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

DIAGNOSIS OF LUNG DISEASES USING PULMONARY SOUNDS

by

Fevzi Yasin KABABULUT

February, 2024

İZMİR

DIAGNOSIS OF LUNG DISEASES USING PULMONARY SOUNDS

A Thesis Submitted to the
Graduate School of Natural and Applied Sciences of Dokuz Eylül University
In Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy in Department of Electrical and Electronics Engineering Program

by
Fevzi Yasin KABABULUT

February, 2024
İZMİR

Ph.D. THESIS EXAMINATION RESULT FORM

We have read the thesis entitled **"DIAGNOSIS OF LUNG DISEASES USING PULMONARY SOUNDS"** completed by **FEVZİ YASİN KABABULUT** under supervision of **ASSOC. PROF. DR. DAMLA GÜRKAN KUNTALP** and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Doctor of Philosophy.

Assoc.Prof.Dr. Damla GÜRKAN KUNTALP

Supervisor

Prof. Dr. Olcay AKAY	Prof. Dr. Adil ALPKOÇAK
Thesis Committee Member	Thesis Committee Member
Assoc. Prof. Dr. Yalçın İŞLER	Assoc. Prof. Dr. Yavuz ŞENOL
Examining Committee Member	Examining Committee Member

Prof. Dr. Okan FISTIKOĞLU
Director
Graduate School of Natural and Applied Sciences

ACKNOWLEDGEMENT

Firstly, I am very grateful to my life partner and wife Ebru Kababulut for her support and encouragement during my graduate education. I would like to extend my heartfelt gratitude to my 5-year-old daughter Esma Zeynep KABABULUT for her understanding and patience in giving up our playtime while I pursued my studies.

Undoubtedly the biggest contributions in making this thesis belong to my advisor Assoc. Prof. Dr. Damla Kuntalp and Prof. Dr. Mehmet Kuntalp. For her valuable review, guidance, and contribution I am very grateful to my advisor Assoc. Prof. Dr. Damla Kuntalp. I am indebted to Prof. Dr. Mehmet Kuntalp for sharing all kinds of information and his valuable opinions with me. I would like to thank Prof. Dr. Olcay Akay, Prof. Dr. Adil Alpkoçak and all members of the Dokuz Eylul University Electrical and Electronics Faculty who have supported me at courses and other times.

I would like to thank Nermin Özcan and Okan Düzyel because of their contribution to my thesis. I also would like to thank Ministry of Transport and Infrastructure İzmir Seabed Dredging Chief Engineer employees for their support during my graduate education.

Finally, of course, I am indebted to all my family members especially my mother and my father for always believing in me and for their support.

Fevzi Yasin KABABULUT

DIAGNOSIS OF LUNG DISEASES USING PULMONARY SOUNDS

ABSTRACT

Millions of people in the world suffer from lung diseases. In this thesis, we aim to develop different techniques to help diagnose lung diseases by analyzing pulmonary sounds. We used the ICBHI 2017 database, which is a public lung sounds database. Since COPD patients' recordings are majority in this database, we created different datasets by applying data visualization, augmentation, and cleaning operations to eliminate class imbalance. We extracted Mel Frequency Cepstral Coefficients (MFCC), Power Spectral Density (PSD), and statistical features from the lung sounds and worked on increasing the performance of our machine learning algorithms by using data sampling/augmentation methods. We made use of the Shapley values in the selection of features in two ways: by considering the absolute Shapley values and by using the contribution of Shapley values on a class basis we created five different feature sets. Using these feature sets, for six classes from the database (Healthy, URTI, COPD, Bronchectasy, Pneumonia and Bronchiolitis), we made sextuple, quintuple, quadruple, triple and binary classifications. We measured the classification performances of our machine learning algorithms on different metrics, some of them using Shapley methods and some without. We also selected the most successful Shapley methods as decision makers and designed ensemble clinical support systems that classified lung diseases based on the results of five binary classifications. Decision Tree algorithm was used in the ensemble systems. The reason for using the decision tree algorithm is that Shapley values, indicating the class contribution of features, can be obtained with decision tree-based algorithms.

Keywords: Machine learning algorithms, U-Matrix, t-SNE, MFCC, PSD, Shapley value, lung diseases classification

AKCİĞER SESLERİ KULLANILARAK AKCİĞER HASTALIKLARI TEŞHİSİ

ÖZ

Dünyada milyonlarca insan akciğer hastalıklarından muzdarip. Bu tezde amacımız akciğer seslerinde yapılacak analizler ile akciğer hastalıkları teşhisine yardımcı olacak farklı teknikler geliştirmektir. Açık erişimli akciğer sesleri veritabanı olan ICBHI 2017 veritabanını kullandık. Bu veritabanında KOAH hasta kayıtları çoğunlukta olduğundan sınıf dengesizliğini ortadan kaldırmak adına veri görüntüleme, arttırma ve temizleme işlemlerini uygulayarak farklı verisetleri oluşturduk. Akciğer seslerinden Mel Frekans Kepstral Katsayıları (MFCC), Güç Spektral Yoğunluğu (PSD) ve istatistiksel öznitelikleri çıkardık ve veri örnekleme/arttırma yöntemlerini kullanarak makine öğrenme algoritmalarımızın performansını arttırmaya çalıştık. Özniteliklerin seçiminde Shapley değerlerinden iki şekilde yararlandık: Mutlak Shapley değerlerini dikkate alarak ve sınıf bazında Shapley değerlerinin katkısını kullanarak beş farklı öznitelik seti oluşturduk. Bu özellik setlerini kullanarak veri tabanındaki altı sınıf için (Sağlıklı, ÜSYE, KOAH, Bronşektazi, Zatüre ve Bronşiolit) altılı, beşli, dörtlü, üçlü ve ikili sınıflandırmalar yaptık. Bunların bir kısmında Shapley metotları kullanarak, bir kısmında ise kullanmadan makine öğrenme algoritmalarımızın farklı metriklerde sınıflandırma performanslarını ölçtük. Ayrıca Shapley metotlarından en başarılı olanlarını karar verici olarak seçip beş adet ikili sınıflandırmadan gelen sonuca göre akciğer hastalığı tahmini yapan topluluk klinik destek sistemleri tasarladık. Topluluk sistemlerinde Karar Ağacı algoritması kullanıldı. Karar ağacı algoritmasının kullanılmasının nedeni, özniteliklerin sınıf katkısını gösteren Shapley değerlerinin karar ağacı tabanlı algoritmalarla elde edilebilmesidir.

Anahtar Kelimeler: Makine öğrenme algoritmaları, U-Matrix, t-SNE, MFCC, PSD, Shapley değeri, akciğer hastalıkları sınıflandırması

CONTENTS

Page

Ph.D. THESIS EXAMINATION RESULT FORM ii
ACKNOWLEDGEMENT iii
ABSTRACT iv
ÖZ v
LIST OF FIGURES xi
LIST OF TABLES xii

CHAPTER ONE-INTRODUCTION 1

1.1 Motivation 1
1.2 Aim 1
1.3 Related Works 2

CHAPTER TWO-PULMONARY SOUNDS AND LUNG DISEASES 14

2.1 Pulmonary Sounds 14
2.1.1 Crackles 14
2.1.2 Wheezes 14
2.1.3 Rhonchi 14
2.1.4 Stridor 15
2.1.5 Squawk 15
2.1.6 Bronchial 15
2.2 Lung Diseases 15
2.2.1 Upper Respiratory Tract Infection (URTI) 16
2.2.2 Lower Respiratory Tract Infection (LRTI) 16
2.2.3 Chronic Obstructive Pulmonary Disease (COPD) 16
2.2.4 Asthma 16
2.2.5 Bronchiectasis 17
2.2.6 Pneumonia 17
2.2.7 Bronchiolitis 17

CHAPTER THREE-DATABASE 18

3.1 ICBHI 2017 Database 18
3.2 Our Datasets 19
3.2.1 Authentic Datasets 19
3.2.1.1 Two-Class Authentic Datasets 19
3.2.1.2 Three-Class Authentic Datasets 22
3.2.1.3 Four-Class Authentic Datasets 23
3.2.1.4 Five-Class Authentic Datasets 23
3.2.1.5 Six-Class Authentic Datasets 24
3.2.2 Synthetic Datasets 24
3.2.2.1 Two-Class Synthetic Datasets 25
3.2.2.2 Three-Class Synthetic Datasets 26
3.2.2.3 Six-Class Synthetic Datasets 26

CHAPTER FOUR-PULMONARY SOUND ANALYSIS METHODS 28

4.1 Features Used in Pulmonary Sound Analysis 28
4.1.1 Mel Frequency Cepstral Coefficients (MFCC) 28
4.1.1.1 Framing 28
4.1.1.2 Windowing 29
4.1.1.3 Discrete Fourier Transform (DFT) 29
4.1.1.4 Mel-Filter Bank 30
4.1.1.5 Discrete Cosine Transform (DCT) 31
4.1.2 Power Spectral Density (PSD) 31
4.1.3 Statistical Features (STF) 32
4.2 Machine Learning Algorithms Used in Pulmonary Sound Analysis 34
4.2.1 Decision Tree 34
4.2.2 Random Forest 35
4.2.3 XGBoost Algorithm 37
4.2.4 K-Nearest Neighbour (K-NN) 38
4.2.5 Support Vector Machine (SVM) 40

4.2.6 Artificial Neural Network (ANN) 42
4.2.7 Logistic Regression 44

CHAPTER FIVE-VISUALIZATION OF DATA DISTRIBUTION..... 45

5.1 The Self Organized Map (SOM) Networks 45
5.1.1 U-Matrix Maps 45
5.1.2 Somhit Maps 46
5.1.3 Interpretation of SOM Network Maps 47
5.2 The t-Distributed Stochastic Neighbor Embedding (t-SNE) Method 53
5.2.1 t-SNE Outputs of 6 Class-Distribution 53
5.2.2 t-SNE Outputs of 5 Class-Distribution 55
5.2.3 t-SNE Outputs of 2 Class-Distribution 57
5.2.4 Interpretation of t-SNE Outputs 59
5.3 Density Based Spatial Clustering of Applications with Noise (Dbscan) 60
5.3.1 Dbscan Outputs of Healthy Class 61
5.3.2 Dbscan Outputs of URTI Class 62
5.3.3 Dbscan Outputs of COPD Class 62
5.3.4 Dbscan Outputs of Pneumonia Class 63
5.3.5 Dbscan Outputs of Bronchiolitis Class 64
5.3.6 Interpretation of Dbscan Outputs 64

CHAPTER SIX-METHODS FOR IMBALANCED DATASETS..... 65

6.1 Under Sampling Methods 65
6.1.1 Random Under Sampling (RUS) Method 65
6.1.2 Nearmiss Method 65
6.1.3 Tomek Method 66
6.2 Over Sampling Methods 66
6.2.1 Random Over Sampling (ROS) Method 66
6.2.2 Smote Method 66
6.3 Classification Results of Sampling Methods 67

6.4 Adaptive Synthetic Sampling Approach (ADASYN) Method 69
6.5 Synthetic Data Generation by Adding Noise 71
6.6 Generative Adversarial Network (GAN) 72

CHAPTER SEVEN-FEATURE SETS CONSTRUCTED USING SHAPLEY VALUES 76

7.1 Shapley Value 76
7.2 Feature Sets 76
7.2.1 Feature Set 1 80
7.2.2 Feature Set 2 80
7.2.3 Feature Sets 3 83
7.2.4 Feature Sets 4 85
7.2.5 Feature Sets 5 87

CHAPTER EIGHT-PROPOSED CLASSIFICATION METHODS AND SIMULATIONS RESULTS 91

8.1 Evaluation Metrics 91
8.2 Binary Classifications 92
8.2.1 Binary Classifications without Shapley Methods 92
8.2.1.1 Healthy Class Binary Classifications 93
8.2.1.2 URTI Class Binary Classifications 94
8.2.1.3 COPD Class Binary Classifications 98
8.2.1.4 Pneumonia Class Binary Classifications 99
8.2.1.5 Bronchiolitis Class Binary Classifications 102
8.2.2 Binary Classifications with Shapley Methods 105
8.2.2.1 Healthy and The Others Classifications 105
8.2.2.2 URTI and The Others Classifications 106
8.2.2.3 COPD and The Others Classifications 107
8.2.2.4 Pneumonia and The Others Classifications 108
8.2.2.5 Bronchiolitis and The Others Classifications 109

8.3 Ensemble Classifications 110
8.4 Three-Class Classifications 111
8.5 Four-Class Classifications 112
8.6 Five-Class Classifications 113
8.7 Six-Class Classifications 114
8.8 Simulations Results 114

CHAPTER NINE-DISCUSSION 122

CHAPTER TEN-CONCLUSION 126

REFERENCES 128

LIST OF FIGURES

Page

Figure 3.1 Hierarchic and non-hierarchic datasets 22
Figure 4.1 Mel-Filter Bank 31
Figure 5.1 U-matrix map of iris flower dataset 46
Figure 5.2 Somhit map of iris flower dataset 46
Figure 5.3 U-matrix of all MFCC features 48
Figure 5.4 Somhit of all MFCC features 48
Figure 5.5 Somhit of healthy MFCC features 48
Figure 5.6 Somhit of unhealthy MFCC features 48
Figure 5.7 Somhit of URTI MFCC features 49
Figure 5.8 Somhit of COPD MFCC features 49
Figure 5.9 Somhit of Bronchiectasy MFCC features 49
Figure 5.10 Somhit of Pneumonia MFCC features 49
Figure 5.11 Somhit of Bronchiolitis MFCC features 49
Figure 5.12 U-matrix of all PSD features 50
Figure 5.13 Somhit of all PSD features 50
Figure 5.14 Somhit of healthy PSD features 50
Figure 5.15 Somhit of unhealthy PSD features 50
Figure 5.16 Somhit of URTI PSD features 50
Figure 5.17 Somhit of COPD PSD features 50
Figure 5.18 Somhit of Bronchiectasy PSD features 51
Figure 5.19 Somhit of Pneumonia PSD features 51
Figure 5.20 Somhit of Bronchiolitis PSD features 51
Figure 5.21 U-matrix of all STF features 51
Figure 5.22 Somhit of all STF features 51
Figure 5.23 Somhit of healthy STF features 52
Figure 5.24 Somhit of URTI STF features 52
Figure 5.25 Somhit of COPD STF features 52
Figure 5.26 Somhit of Bronchiectasy STF features 52
Figure 5.27 Somhit of Pneumonia STF features 52

LIST OF FIGURES

Page

Figure 5.28 Somhit of Bronchiolitis STF features 52
Figure 5.29 t-SNE of 6 class MFCC features 54
Figure 5.30 t-SNE of 6 class PSD features 54
Figure 5.31 t-SNE of 6 class STF features 55
Figure 5.32 t-SNE of 5-class MFCC features 56
Figure 5.33 t-SNE of 5-Class PSD features 56
Figure 5.34 t-SNE of 5-class STF features 57
Figure 5.35 t-SNE of 2 Class MFCC features 58
Figure 5.36 t-SNE of 2 Class PSD features 58
Figure 5.37 t-SNE of 2 Class STF features 59
Figure 5.38 Dbscan of healthy class MFCC features 61
Figure 5.39 Dbscan of URTI class MFCC features 62
Figure 5.40 Dbscan of COPD class MFCC features 63
Figure 5.41 Dbscan of Pneumonia class MFCC features 63
Figure 5.42 Dbscan of Bronchiolitis class MFCC features 64
Figure 7.1 Ranking of all 68 features according to their Shapley values for all classes 77
Figure 7.2 Ranking of most contributing 30 features according to their Shapley values for 2-class (Healthy and the Others) classification problem 78
Figure 7.3 Ranking of most contributing 30 features according to their Shapley values for 2-class (URTI and the Others) classification problem 78
Figure 7.4 Ranking of most contributing 30 features according to their Shapley values for 2-class (COPD and the Others) classification problem 79
Figure 7.5 Ranking of most contributing 30 features according to their Shapley values for 2-class (Pneumonia and the Others) classification problem 79
Figure 7.6 Ranking of most contributing 30 features according to their Shapley values for 2-class (Bronchiolitis and the Others) classification problem 80
Figure 8.1 Ensemble System (Method 3) 110

LIST OF FIGURES

Page

Figure 8.2 Ensemble System (Method 4)..... 111
Figure 8.3 Processes of Shapley methods..... 121

LIST OF TABLES

Page

Table 3.1 Distribution of people from whom pulmonary sound recordings were obtained based on diagnosis 18
Table 3.2 Class distribution in two-class authentic datasets 20
Table 3.3 Class distribution in three-class authentic datasets 23
Table 3.4 Class distribution in four-class authentic datasets 23
Table 3.5 Class distribution in five-class authentic datasets 24
Table 3.6 Class distribution in six-class authentic datasets 24
Table 3.7 Class distribution in two-class synthetic datasets 25
Table 3.8 Class distribution in three-class synthetic datasets 26
Table 3.9 Class distribution in six-class synthetic datasets 27
Table 6.1 Performance values of imbalanced methods 67
Table 6.2 Performance values of algorithms using dataset 1 69
Table 6.3 Performance values of algorithms using dataset 2 71
Table 6.4 Performance values of algorithms using dataset 3 72
Table 6.5 Performance values of Random Forest algorithm using GAN datasets 73
Table 6.6 Performance values of Random Forest algorithm using dataset 12 and authentic dataset 74
Table 7.1 Absolute Shapley values of feature set 2 81
Table 7.2 Shapley values of feature sets 3 on class basis 83
Table 7.3 Absolute Shapley values of feature sets 4 in binary classifications 85
Table 7.4 Percentage of Shapley values of feature sets 5 88
Table 8.1 Performance values of Random Forest algorithm using binary healthy class datasets 93
Table 8.2 Performance values of machine learning algorithms using dataset 16 93
Table 8.3 Performance values of machine learning algorithms using binary URTI class datasets 95
Table 8.4 Performance values of machine learning algorithms using binary COPD class datasets 99

LIST OF TABLES

Page

Table 8.5 Performance values of machine learning algorithms using binary Pneumonia class datasets 99
Table 8.6 Performance values of machine learning algorithms using binary Bronchiolit class datasets 102
Table 8.7 5-fold cross-validation performance of Decision Tree algorithm for dataset 43 106
Table 8.8 5-fold cross-validation performance of Decision Tree algorithm for dataset 42 106
Table 8.9 5-fold cross-validation performance of Decision Tree algorithm for dataset 31 107
Table 8.10 5-fold cross-validation performance of Decision Tree algorithm for dataset 41 108
Table 8.11 5-fold cross-validation performance of Decision Tree algorithm for dataset 40 109
Table 8.12 Performance values of Random Forest algorithm using three-class datasets 111
Table 8.13 Performance values of Random Forest algorithm using dataset 19 113
Table 8.14 Performance values of machine learning algorithms using five-class datasets 113
Table 8.15 Performance values of machine learning algorithms using dataset 21 114
Table 8.16 Comparison of the performances of different methods 118

CHAPTER ONE
INTRODUCTION

1.1 Motivation

The Covid 19 pandemic has shown that respiratory viruses and bacteria can infect people very quickly and threaten people's vital functions very easily. Therefore, early diagnosis and treatment of respiratory diseases will save lives. With the use of effective expert machine learning algorithms, automatic diagnosis can be made in a much faster fashion or doctors can be assisted in their diognosis process by providing reliable pre diagnosis results.

1.2 Aim

Respiratory diseases are divided into two groups: upper respiratory diseases and lower respiratory diseases. In this thesis, we will deal with lower respiratory tract diseases, which are also characterized as lung diseases.

This study aims to detect whether people have lung disease or which lung disease they have through machine learning algorithms using their pulmonary sounds. In this thesis, we tried to answer the following research questions. How can the separability of the dataset be observed in terms of different features? How to make an unbalanced data set balanced? What kind of methods are used to achieve more successful classification in imbalanced datasets? How much can classification performance be increased with these methods? Which features are important in diagnosing lung diseases and how are they selected? How is lung disease classification done using one-dimensional data? Are there any methods to increase classifier performance other than traditional methods in feature selection?

In the following chapter, we will briefly describe lung sounds and lung diseases. In the third chapter, we will examine the ICBHI 2017 database, which is the source of the lung sounds we used in our thesis, and the datasets we derived from this database. In the fourth chapter, we will explain the features and machine learning algorithms

that are widely used in lung sound analysis in the literature. In the fifth chapter, we will discuss class-based distribution of our data using different visualization methods. In the sixth chapter, we will share the different methods and results we used to make the class distributions more balanced in imbalanced datasets. In the seventh chapter, we will explain the Shapley values that we use for both dimensionality reduction, i.e. feature selection, and improvement of the performance of our algorithms, and the new feature sets that we obtain by different arrangement of the features according to the Shapley values. In the eighth chapter, we will analyze our binary and multiclass classification algorithms, the performance metrics we used and our simulation results. In the last chapter, we will discuss our results in detail and talk about possible future studies.

1.3 Related Works

A stethoscope is a device used to listen to the internal sounds of the body that doctors utilize for the diagnosis of diseases. In contrast to many diagnostic methods, listening with a stethoscope (auscultation) is a non-invasive method that is very easy to apply and also cause no harmful effects to the patient. With this method, many internal organ sounds, especially heart and lung sounds, can be listened and interpreted depending on the experience and ability of the doctor for diagnosing the diseases. However, auscultation has some important disadvantages too. Correct diagnosis depends heavily on the doctor's experience, ability, and also on ambient noise present. Digital stethoscopes convert the sounds heard during auscultation to electrical signals and then to the digital data that can be recorded. The sounds recorded with a digital stethoscope can be processed and analyzed and usefull results can be obtained by means of digital signal processing methods which were used by many studies in the literature, and that way may assist doctors in diagnosis process.

Heart sounds are the leading sounds that are classified by signal processing methods. Because of the prevalence of the diseases in the world and being among the mortal diseases, researchers have often focused on heart diseases, and therefore heart sounds. While the most common methods used by the researchers for heart sound

segmentation are the Hidden Semi-Markov Model (HSMM) (Oliveira et al., 2019), Ensemble Empirical Mode Decomposition (EEMD) (Papadaniil & Hadjileontiadis, 2014) and Shannon's Energy Based Method (Nigam & Priemer, 2005), the methods used for the classification of heart sounds are Convolutional Neural Network (CNN) (Nilanon et al., 2016; Potes et al., 2016; Rubin et al., 2016; Y. Zhang et al., 2016), Artificial Neural Network (ANN) (Clifford et al., 2016; Dwivedi et al., 2019; Gupta et al., 2007; Liu et al., 2016; Martínez-Alajarín et al., 2007; Olmez & Dokur, 2003), Support Vector Machine (SVM) (Dwivedi et al., 2019; Y. Zhang et al., 2016), Hidden Markov Model (HMM) (Zhong et al., 2013), K Nearest Neighbor (KNN) (Randhawa & Singh, 2015).

Lung sound is another internal organ sound that is the main theme of our thesis and is also widely studied. Fast Fourier Transform (FFT) (Jácome et al., 2019; Moussavi et al., 1998) and CNN are used together for the segmentation of lung sounds, while machine learning algorithms such as KNN (Amaral et al., 2012; C. H. Chen et al., 2015; Grønnesby et al., 2017; Gurung et al., 2011; Palaniappan et al., 2014; Rao et al., 2019; Rocha et al., 2007), SVM (Amaral et al., 2012; Grønnesby et al., 2017; Palaniappan et al., 2014), ANN (Gurung et al., 2011; Rao et al., 2019; Rocha et al., 2007) and CNN (Abdoli et al., 2019; Aykanat et al., 2017; Bardou et al., 2018; Chatterjee et al., 2019; Liang & Hu, 2015) which are also used in heart sounds, are preferred for classification of lung sounds.

In the literature, there are many studies that classifies patients' respiratory sounds by machine learning algorithms. When we look at these studies, it is seen that the algorithms developed either classify the abnormal sounds such as wheeze and crackle, which are the symptoms of lung diseases, or classify the common lung diseases such as asthma, pneumonia and COPD. Unhealthy respiratory sounds contain sounds with distinctive characteristics such as wheeze, crackle, rhonchi, stridor, squawks, and bronchial depending on type of the disease. The studies in the first group focus only on the classification of these abnormal sounds (Bardou et al., 2018; Chatterjee et al., 2019; Grønnesby et al., 2017; Sengupta et al., 2016; Waitman et al., 2000). In the second group, the studies concentrate on binary classification to determine whether the

person has a certain lung disease (asthma, pneumonia or COPD) or not (Badnjevic et al., 2015; Chatterjee et al., 2019; Fraiwan et al., 2021; García-Ordás et al., 2020; Habukawa et al., 2017; Haider & Behera, 2022; Heitmann et al., 2023; Naqvi & Choudhry, 2020) .

In a study (Pham et al., 2020) using the same dataset as ours, 3 different algorithms (CNN Mixture of Expert-CNN MOE, CRNN and their ensemble) classified lung sounds into 4 classes as crackle, wheeze, crackle & wheeze and normal in Task 1.1, into 2 classes as normal and abnormal cycles in Task 1.2. The same algorithms classified lung diseases into 3 classes as healthy, chronic and non-chronic in Task 2.1, and into 2 classes as healthy and unhealthy in Task 2.2. The performance of the algorithms was evaluated using different cycle lengths ranging from 2 sec to 8 sec and different time patch widths ranging from 0.6 sec to 3 sec. The ensemble algorithm, which used cycle lengths of 6 second giving the best results for Task 1, was the most successful algorithm with 0.89 specitivity and 0.72 sensitivity scores in Task 1.1, 0.89 specitivity and 0.82 sensitivity scores in Task 1.2. The CNN-MOE algorithm, which used patch widths of 2.4 seconds giving the best results for Task 2, was the most successful algorithm with 0.86 specitivity and 0.96 sensitivity scores in Task 2.1, and 0.86 specitivity and 0.99 sensitivity scores in Task 2.2 (Pham et al., 2020).

In another study (Engin et al., 2022), using machine learning algorithms such as LDA, KNN, SVM, and NaiveBayes, lung sounds were classified as normal, rhonchi, fine crackle and coarse crackle. In algorithms, 11 different feature extraction methods were used, including Mel Frequency Cepstral Coefficients (MFCC), time domain, frequency domain and linear predictive coding (LPC) features. In this study, lung sound recordings containing a total of 400 respiratory cycles from 94 different people were used. The KNN algorithm correctly classified these recordings into four lung sounds with 90.63% accuracy and 94.00% Area under the ROC Curve (AUC) scores using the mean and standard deviation of the MFCC feature and the standard deviation of the LPC feature (Engin et al., 2022).

In a survey study (Tabatabaei et al., 2021), studies that performed cough detection, snore detection, crackle detection, wheeze detection and sleep monitoring from lung sounds via smartphones were examined. The methods and performances of the studies which made crackle and wheeze detection by an acoustic sensor and algorithms used in the smartphone, were mentioned. One of these mentioned studies (Olvera-Montes et al., 2018) performed crackle detection based on sudden changes in the coefficients of the time varying autoregressive model in sound data sampled at 10 kHz, and showed a performance of 94% in accuracy and 94.4% in sensitivity. In the other mentioned study (Oletic & Bilas, 2016), 19 asthmatic wheeze sounds and 18 healthy sounds sensed by the wearable acoustic sensor were detected by the smartphone using the signal reconstruction method with 80% accuracy.

In a study, using lung sound data, 7 lung sounds (normal, roughness, coarse crackling, monophonic wheeze, polyphonic wheeze, stridor, and squawk) were classified with improved-random forest, adaboost and gradient boosting algorithms (Jaber et al., 2020). From the R.A.L.E. (Respiration Acoustics Laboratory Environment) lung sounds database, 15 appropriate lung sound recordings containing 7 lung sounds were used as a data set in this study. 4 statistical features extracted using the 4th order Daubechies wavelet and 12 MFCC features were used as input data. The accuracies of improved-random forest, adaboost and gradient boosting algorithms using the MFCC features are 98.76%, 96.29%, and 94.71%, respectively. The accuracies of improved-random forest, adaboost and gradient boosting algorithms using Daubechies wavelet features are 99.04%, 96.63% and 95.11%, respectively (Jaber et al., 2020).

In another study, COPD, Pneumonia and normal lung sounds are classified by machine learning algorithms that use 116 features as input data, extracted from lung sound recordings in the ICHBI 2017 database (Naqvi & Choudhry, 2020). Of the 116 features, 19 are in the time domain, 12 are in the frequency domain, 26 are in the Cepstral domain, and 59 are texture-based features. The most successful of the 22 algorithms using the new feature which is created by selecting 25 of them and

combining, is the Quadratic Discriminant algorithm with 99.70% accuracy (Naqvi & Choudhry, 2020).

In another study, the waveforms and peak points of the sound recordings of lung patients of different ages and genders were determined and diseases estimations were made (Jindal et al., 2018). First of all, waveforms in the frequency domain were extracted using the Welch method for lung sounds known to be wheeze, crackle and stridor, and the points where these forms were at global maximum were determined. By looking at the frequency ranges of these points, the diseases of people of different ages were successfully estimated according to this study (Jindal et al., 2018).

In other study, features of lung sounds such as RMS, Zero Crossing, Turn Count, Mean, Variance and Form Factor were extracted using Discrete Fourier Transform (DFT) and FFT (Manir et al., 2020). Firstly, the envelope and peak points of 10 abnormal lung sounds and 1 normal lung sound in the time domain and Burg Power Spectral Density (PSD) estimate in the frequency domain were plotted. Although the features of sound recordings extracted using these graphs are different for each lung sound, it was stated that more data is needed to use these features in the detection of lung diseases, according to the study (Manir et al., 2020).

In a study, in Covid 19 cases, intensive care unit admission (ICUA) and death of the patients were estimated using Logistic Regression and Gradient Boosted Decision Tree (GBDT) algorithms according to the Shapley Values of the patient's demographic and comorbidity information (Cavallaro et al., 2021a). Firstly, the Odd Ratio (OR) values of 37 pieces of ethnicity, gender and comorbidity information of the patients were calculated using the Logistic Regression model. Then, using the XGBoost library, the Shapley Values of these informations of the patients were calculated by the GBDT model. Finally, the Receiver Operating Characteristic (ROC)-AUC prediction values of the GBDT model using these calculated values are 0.83 for ICUA, 0.68 for death, while the prediction values of the Logistic Regression model are 0.80 for ICUA and 0.68 for death (Cavallaro et al., 2021a).

In another study, using lung sound data, normal, crackle and wheeze sounds were classified with kNN, SVM, Naive Bayes, Decision Tree and Random Forest algorithms (Güler et al., 2020). In this study, 20 normal, 20 crackle and 20 wheeze lung sounds data and 60 normal, 60 crackle and 60 wheeze lung sounds data, which were augmented with artificial data production, were used as two different datasets. The best 4 of MFCC based features of the 36 ones extracted by the Librosa library, and the best 5 of MFCC based features of the 200 ones extracted by the Jaudio software were used as input data. Decision tree and random forest algorithms, which were the most successful algorithms using these features, made 94.00% correct predictions. Also, in the 10 fold cross verification results, the Naive Bayes algorithm was the most successful classifier with 95.00% (Güler et al., 2020).

In two studies where abnormal respiratory sounds were classified by Convolutional Neural Networks (CNN), the spectrograms of recorded sounds were used as input data. In the first study, classifications were made by using 4 different dataset (Aykanat et al., 2017). In the first dataset, respiratory sounds are classified into two classes as normal and pathological. In the second dataset, respiratory sounds are classified into 13 classes as normal, rhonchus, squawk, stridor, wheeze, rales, bronchovesicular, friction rub, bronchial, absent, decreased, aggravation, or long expirium duration (LED). In the third dataset, respiratory sounds are classified into three classes as normal, rale, and rhonchus. In the fourth dataset, respiratory sounds were classified into a total of 78 classes, including all classes with more than one label. In this study, the MFCC's of the recorded sounds were used as the input data for the SVM classifier, while the spectrograms of the sounds were used as the input data for the CNN classifier. It was observed that the accuracy results were 86% in CNN and 86% in SVM for the first dataset, were 76% in CNN and 75% in SVM for the second dataset, were 80% in CNN and 80% in SVM for the third dataset and were 62% in CNN and 62% in SVM for the fourth dataset (Aykanat et al., 2017).

The second study made classifications with 3 different approaches (Bardou et al., 2018). The performance of K-NN, SVM, GMM and CNN classifiers which used 6 MFCC statistical features, were compared in the first approach. The performances of

the K-NN, SVM, GMM and CNN classifiers which used the Local Binary Pattern (LBP is an operator that labels each pixel according to a threshold value by taking into account the neighborhood of pixels and assigns a binary value) extracted from the spectrograms of lung sounds, were compared in the second approach. In the third approach, the performance of the CNN classifier was tried to be improved by different techniques using only spectrograms of lung sounds. In the first approach, the most successful classifier was CNN with 91.67% accuracy, followed by SVM with 91.12% accuracy. In the second approach, the most successful classifier was CNN with 80.00% accuracy, followed by SVM with 71.21% accuracy. In the third approach, the CNN classifier first achieved 93.60% accuracy and then increased the performance to 95.10% using the data augmentation method. Finally, the softmax activation outputs of the 4 most successful CNN models with different iterations were summed up and accuracy performance was increased to 95.56% (Bardou et al., 2018).

In the next reference (Bahoura & Pelletier, 2004), classification of respiratory sounds as normal or wheezing was done by Gaussian Mixture Model (GMM) classifier by using Mel Frequency Cepstral Coefficients (MFCC) as the feature extraction method. Results are compared with of Subband-based Cepstral (SBC) method as the feature extraction method and Multilayer Perceptron (MLP) Neural Network and Vector Quantization (VQ) methods as the classifiers. At the end of the study, the lowest standard error was obtained with MFCC and GMM methods (Bahoura & Pelletier, 2004).

Three studies use different feature extraction and classification methods to detect wheezing of respiratory sounds. The first of these studies used Fourier Transform (Spectrogram) as the feature extraction method and used ANNs as the classifier (Rietveld et al., 1999). In this study, spectrograms are interpreted and classiffied into three classes. Then the vectors obtained from the spectrogram are given as inputs and classified by unsupervised learning neural network. Classification results were subsequently labeled as severe asthma, mild asthma, and not asthma classes, and a comparison was made between these two methods. Although both methods could not clearly classify spectrogram data into three classes, the feed-forward neural network,

which is used as a third method, was able to classify training data with 95% success and test data with 43% success (Rietveld et al., 1999).

The second study also used the Fourier Transform as the feature extraction method, and used Vibration Response Imaging (VRI) as the classifier (Guntupalli et al., 2008). The VRI system used in this study consists of a device that records lung sounds by means of 2 inactive and 40 active piezoelectric sensors placed in the back chest. The automatic wheeze detection (AWD) algorithm is a software that detects wheeze sounds using spectral characteristics. The performance analysis of the AWD algorithm was performed by considering the wheeze sound in lung sounds of 3 chest specialists. The VRI AWD algorithm provides 83% accuracy; 85% specificity; 88% negative predictive value (NPV); 79% positive predictive value (PPV) (Guntupalli et al., 2008).

The third study used the Short Time Fourier Transform (STFT) as the feature extraction method and used artificial neural networks as the classifier (Riella et al., 2009). In this study, the spectrogram of lung sounds was extracted with STFT. The obtained spectrograms were passed through a Laplace mask, which is one of the digital image processing methods and then low frequencies are eliminated and high frequencies are isolated with an acceptable threshold value. The isolated high frequencies are converted into a series with their magnitudes and given as input to the Multilayer Perceptron (MLP) Neural Network. The 112 respiratory cycles given as input were classified into two classes as wheeze sound or no sound at the output of the neural network. The overall classification accuracy of this study is 84.82% (Riella et al., 2009).

Two studies use different feature extraction and classification methods for crackle detection of respiratory sounds. The first of these studies used AR model coefficients, crackle parameters and Wavelet Transform (WT) for feature extraction, while K-NN and back-propagation neural network methods were used as the classifier (Kahya et al., 2006). Among the feature extraction methods used in this study, the 6th degree AR model coefficients were calculated by dividing the respiratory cycle into 30 segments with 25% overlapping and representing each segment with a vector of 6 coefficients.

Crackle parameters were calculated by looking at the duration of the first deviation in each segment and the duration of the first two cycles together with the determination of crackle sound. Wavelet coefficients were calculated using 8th order Daubechies wavelet. These features were given to K-NN and artificial neural network classifiers and classified as diseased sound or healthy sound. When the artificial neural network made classification using crackle parameters and AR model coefficients, it performed better than K-NN with 92.5% accuracy, 90% specificity, 95% sensitivity performance at inhalation and 95% accuracy, 95% specificity, 90 sensitivity performance at exhalation (Kahya et al., 2006).

The second study attempted to diagnose interstitial pneumonia (IP) disease using FT (Ono et al., 2009). In this study, Average Linear Intensity (ALI) was calculated by passing lung sounds which were recorded by phonopnemography, through Hanning window and then exposing to Fast Fourier Transform (FFT). Four frequency reference indices were extracted from the calculated ALI graph. In addition, fine crackle detection was performed from lung sounds listened by 2 chest specialists. Frequency indices, fine crackle sounds and other lung tests obtained from the patients were also evaluated and some inferences were made about the diagnosis and severity of interstitial pneumonia diseases. As a result of this study, it was concluded by the researchers that spectral analysis of lung sounds performed by Fast Fourier Transform (FFT) was successful (Ono et al., 2009).

Two studies use different feature extraction and classification methods for normal, wheezing and crackle detection of respiratory sounds. In the first of these studies, Time Expanded Waveform Analysis (TEWA) was used for feature extraction, while statistical evaluation was performed to diagnose patients which have pnemonia (Murphy et al., 2004). In this study, the multichannel lung sound analyzer, placed in the patient and control groups, automatically determined how much wheeze, crackle and rhoncus the recorded sounds contained in the inhalation and exhalation processes by TEWA. After this determination, according to the number of crackle and rhoncus (the wheeze number was not evaluated), the patients were given a score called pneumonia score. In this study, the success of classification of patients with pneumonia

score higher than 6 was 90% and 78% according to sensitivity criterion and 94% and 88% according to specificity criterion in training and test data, respectively (Murphy et al., 2004).

The second study used Vibration Response Imaging (VRI) for feature extraction, while one chest specialist and one general practitioner classified normal and abnormal images using acquired images (Mor et al., 2007). In this study, computer-assisted multi-sensor respiratory sounds mapping device (VRI) obtained images that were accepted as maps of respiratory sounds from 20 pneumonia patients, 20 pleural effusion patients and 60 healthy people. These two doctors used the images of 20 healthy people for training purposes by considering the differences in the images. Afterwards, they determined normal and abnormal images by evaluating only the remaining images and achieved 82.5% and 80% success according to the sensitivity and specificity criteria, respectively. Finally, in the classification by obtaining additional information such as the presence of abnormal sounds (crackle, wheeze, etc.) and acute symptoms in the patients, they achieved 90% and 88% success according to sensitivity and specificity criteria, respectively (Mor et al., 2007).

In one study, respiratory sounds were classified to three classes as normal, airway obstruction and parenchymal pathology. In this study, the MFCC are used for feature extraction and Support Vector Machine (SVM) is used as the classifier (Palaniappan & Sundaraj, 2013). In the performance evaluation using Confusion Matrix, normal, airway obstruction and parenchymal pathology classification successes were 94.11%, 92.31% and 88%, respectively (Palaniappan & Sundaraj, 2013).

In another study, hybrid scalogram-based features were extracted from lung sounds and lung diseases were classified using a Convolutional Neural Network (CNN) (Shuvo et al., 2021). Continuous Wavelet Transform (CWT) of lung sound recordings taken from International Conference on Biomedical and Health Informatics (ICBHI 2017) database were computed using Morse mother wavelet. Also, Intrinsic Mode Functions (IMF) were computed by taking the Empirical Mode Decomposition of these records. Scalograms with 224x224 resolution were obtained from these wavelet

coefficients and IMFs. The CNN algorithm that made 3-class classification (chronic/ non-chronic/ healthy) achieved 98.92% accuracy by using these scalograms as input data, while the CNN algorithm that made 6-class pathological classification (Bronchiectasis, Bronchiolitis, COPD, Healthy, Pneumonia and URTI) achieved 98.70% accuracy (Shuvo et al., 2021).

Classification of lung sounds as healthy and unhealthy was studied with a Multilayer Perceptron (MLP) artificial neural network using Linear Predictive Cepstral Coefficient (LPCC) based features in (Mukherjee et al., 2021). In ICBHI public dataset, 3642 sound recordings that do not contain crackle and wheeze sounds with a total of 6898 cycles were labeled as healthy, while 3256 were labeled as unhealthy. The MLP neural network using 30-dimensional LPCC feature showed the best performance (with 99.22% accuracy and AUC = 0.9993) when it was compared with BayesNet, SVM, RNN, Naive Bayes, RBF network, Decision Table, LibLINEAR, and Simple logistic algorithms (Mukherjee et al., 2021).

Abnormal lung sounds such as wheeze and crackle were classified using LDA, SVMrbf, RUSBoost, and CNN machine learning algorithms (Rocha et al., 2021). 920 lung sounds from ICBHI database were resaved with fixed (50ms and 100ms) and variable durations (less than 100ms and between 100ms and 2s) and 25 spectral features, 26 MFCC features, and 30 melodic features were extracted. While lung sounds with fixed duration were classified as wheeze, crackle, and other (3 classes) by the CNN algorithm with 96.9% accuracy, in the variable duration scenario lung sounds were classified into 3 classes by the CNN algorithm with 81.8% accuracy (Rocha et al., 2021).

In (García-Ordás et al., 2020) using the ICBHI dataset, lung diseases were estimated using two different classification methods. In the first method, lung diseases were classified as Chronic (COPD, Bronchiectasis, and Asthma), Non-Chronic (Bronchiolitis, Pneumonia, URTI and LRTI) and Healthy using the CNN algorithm. In the second method, five diseases (COPD, Bronchiectasis, Bronchiolitis, Pneumonia and URTI) and healthy class were separated by CNN algorithm. But, due to the

unbalanced distribution of the dataset, synthetic data was created with the Convolutional Variational Autoencoder (VAE) and the dataset was balanced. Mel Spectrograms, a visual representation of the spectrum of lung sounds, were then extracted for the CNN input data. In the last part, for both classifications, the VAE method gave the best results when compared to the other data augmentation methods, Smote and ADASYN. In the first classification, the CNN algorithm performed 0.991 in sensitivity criterion and 0.994 in specificity. In the second classification, the same CNN structure showed a performance of 0.988 in sensitivity criteria and 0.986 in specificity (García-Ordás et al., 2020).

Saldanha J., Chakraborty S., Patil S., Kotecha K., Kumar S., and Nayyar A. used the ICBHI dataset focused on data augmentation with 3 different Variational Autoencoder methods in addition to 7-class classification (Saldanha et al., 2022). Since the dataset is unbalanced in this study, synthetic data for minority classes were produced with Multilayer Perceptron VAE, Convolutional VAE and Conditional VAE methods. 5 deep learning algorithms (Multilayer Perceptron, CNN, Recurrent Neural Network-LSTM, Resnet-50 and Efficient Net B0) try to separate Bronchiectasis, Bronchiolitis, COPD, URTI, Pneumonia, LRTI and healthy classes using Mel-Spectrograms. It has been observed that MLP-VAE and CNN-VAE increase the performance of algorithms more than Conditional VAE. In the classification results, ICBHI scores of Resnet-50, CNN, Efficient Net B0, MLP and LSTM algorithms were 0.85, 0.79, 0.76, 0.74 and 0.67, respectively (Saldanha et al., 2022).

In (Srivastava et al., 2021) using the ICBHI dataset, COPD detection was made from lung sounds using the CNN algorithm. In this study, Mel-Frequency Cepstral Coefficients (MFCC), Mel spectrogram (Mel-Spectrogram), Chromogram calculated from the waveform/power spectrogram of audio recordings (chroma_stft), Constant-Q Chromogram (chroma-cqt) and chroma-cens (Chroma Energy Normalized Variant (CENS)) features are used. The MFCC feature was the one that gave the most successful results in COPD detection. The CNN structure used in this study was compared with some previous studies and it was the most successful algorithm with an ICBHI score of 0.93 (Srivastava et al., 2021).

CHAPTER TWO
PULMONARY SOUNDS AND LUNG DISEASES

2.1 Pulmonary Sounds

Respiratory sounds are divided into two as normal and abnormal sounds. Normal respiratory sounds are also divided as tracheobranchial and vesicular sounds. Tracheobranchial sounds occur in the trachea and in the larger bronchial airways. Vesicular sounds occur in small branches of the respiratory airways. There are six different types of abnormal sounds, and they often indicate a lung disease.

2.1.1 Crackles

Crackles are usually less than 100 ms, usually between 60 and 2000 Hz, short, discontinuous and non-stationary sounds (Pham et al., 2020; Rao et al., 2019). The presence of these sounds indicates diseases such as COPD, pneumonia, fibrosis and bronchiectasis (Rao et al., 2019).

2.1.2 Wheezes

Wheezes are continuous sounds, longer than 250 ms, usually between 100 and 2000 Hz and resembling a sinusoidal signal with high harmonics above 1 kHz at the end of the inspiration process and at the beginning of the expiration process (Guntupalli et al., 2008; Rao et al., 2019). The presence of these sounds indicates diseases such as asthma and COPD (Rao et al., 2019).

2.1.3 Rhonchi

Rhonchi are musical low pitch sounds, usually longer than 100 ms, less than 300 Hz and resembling a sinusoidal signal, such as a wheeze (Rao et al., 2019). The presence of these sounds indicates diseases such as bronchitis and COPD in which mucosa, edema and bronchospasm are concentrated (Rao et al., 2019).

2.1.4 Stridor

Stridor are continuous, over 500 Hz and monophonic sounds (Rao et al., 2019). They have sinusoidal waveform and they are often accepted as a type of wheezing (Rao et al., 2019). The presence of these sounds indicates upper airway obstruction disease (Rao et al., 2019).

2.1.5 Squawks

Squawks are mixed sounds in which a sound resembling wheeze is coming after a crackle, is between about 90-320 ms (Rao et al., 2019). The presence of these sounds indicates fibratic pulmonary disorders (Rao et al., 2019).

2.1.6 Bronchial

Bronchial sounds are characterized as a tubular sound, typically falling within the frequency range of 600-1000 Hz (Rao et al., 2019). In healthy individuals, bronchial sounds are normally heard over the trachea and large bronchi (Rao et al., 2019). However, if these sounds are heard in areas of the lungs where they shouldn't be, it may suggest conditions like consolidation, which can be caused by a lung disease (Rao et al., 2019).

2.2 Lung Diseases

Lung diseases are among the most common diseases in the world because of the prevalence of smoking and air pollution. When you take into account occupational workers in the environment containing harmful gases and passive smokers (non-smokers) who are affected by smoking, we can understand why there are millions of people in the world suffering from these diseases. In this thesis, we will classify the major lung diseases. Therefore, it is important to get enough information about these lung diseases.

2.2.1 Upper Respiratory Tract Infection (URTI)

Upper Respiratory Tract Infection (URTI) is an illness that occurs as a result of inflammation in the upper respiratory areas of the body, such as the nose, throat and sinuses. This disease is often caused by viruses such as rhinovirus, influenza, coronavirus and adenovirus, as well as bacteria such as strep (Florence Nightingale, 2023). It is often confused with flu and cold due to its symptoms such as nasal congestion, runny nose, cough, sneezing and sore throat (Florence Nightingale, 2023).

2.2.2 Lower Respiratory Tract Infection (LRTI)

Lower Respiratory Tract Infection (LRTI) is an illness caused by inflammation in the lower respiratory areas of the body, such as the trachea, bronchi, and bronchioles. This disease is mostly caused by viruses such as RSV, rhinovirus, parainfluenza, influenza adenovirus, as well as bacteria such as S. Pneumonia, H. Influenzae, Mycoplasma pneumonia and B. pertusis (Akşit, 2002). Crowded population, malnutrition, exposure to cigarette smoke, and congenital heart and lung diseases are the main factors that increase the frequency of LRTI (Akşit, 2002).

2.2.3 Chronic Obstructive Pulmonary Disease (COPD)

In Chronic Obstructive Pulmonary Disease (COPD), especially cigarette smoke and other harmful gases and particles cause inflammation in the respiratory airways and bronchi. The destruction of the alveoli due to this inflammation causes complaints such as respiratory difficulties, coughing and shortness of breath (Çiftçi, 2023). COPD is a chronic disease and in a patient with COPD, oxygen is delivered to the tissues in a limited amount since the air is not absorbed sufficiently from the lungs (Çiftçi, 2023).

2.2.4 Asthma

Asthma is a lung disease that makes it difficult for the patient to breathe as a result of narrowing of the airways and can continue for a lifetime with attacks (Memorial Hospitals, 2023). In this disease, air flow is prevented due to swelling of the mucous

membranes in the respiratory tract and the patient has difficulty at breathing (Memorial Hospitals, 2023). This situation is called asthma attacks. In people with asthma, factors such as house dust mites, pollen, cat-dog allergens, domestic fungi, indoor and outdoor pollution trigger asthma attacks (Memorial Hospitals, 2023).

2.2.5 Bronchiectasis

Bronchiectasis is the permanent enlargement that occurs after damage to the bronchi (Polat, 2022). The main causes of this disease include lung infections, airway obstructions, disorders in the immune system and fibrosis in the lung tissue (Polat, 2022). Chronic cough, abundant sputum and shortness of breath are the main symptoms (Polat, 2022).

2.2.6 Pnemonia

Pneumonia is one of the most common cause of death in the world affecting approximately 450 million people and 4 million of them are fatal. It is caused by inflammation of the lung tissue and especially the alveoli. It often occurs at people with reduced body defense when various microorganisms, especially bacteria in mouth, throat or digestive tract, reach the lung (Maltepe Hospital, 2023).

2.2.7 Bronchiolitis

Bronchiolitis is a lung disease that occurs mostly in children under the age of 2 and occurs as a result of inflammation of the bronchioles (Karakış, 2020). Bronchiolitis is often caused by viruses such as RSV, rhinovirus, parainfluenza, influenza virus and adenovirus (Karakış, 2020). Bronchiolitis may begin with runny nose, mild cough, and nasal congestion and continue with symptoms such as shortness of breath, increased heart rate, and restlessness (Karakış, 2020).

CHAPTER THREE
DATABASE

3.1 ICBHI 2017 Database

The International Conference on Biomedical and Health Informatics (ICBHI) 2017 Lung Sounds Database is a public data set of lung sounds created with the contributions of the Universities of Coimbra and Aveiro from Portugal and the University of Thessaloniki from Greece. This dataset includes 920 lung sounds recordings from 126 people duration ranging from 10 sec to 90 sec. These recordings last 5.5 hours in total and include 6895 respiratory cycles. 1864 respirotary cycles include crackles, 882 respirotary cycles include wheezes and 506 respirotary cycles include both crackles and wheezes. Since the number of asthma lung sound recordings was 1 and we could not separate this class data into train and test, we removed this recording from the database. As shown in Table 3.1, 125 people from whom these lung sounds were obtained, 26 of them are healthy people and 99 of them are patients with 6 different lung diseases. Half of the healthy sounds belongs to women. The average age of people with healthy pulmonary sounds is 6.20, the standard deviation is 5.18. 32.32% of unhealty sounds belongs to female patients and 66.67% of them belongs to male patients. The gender of 1.01% is unknown. The average age of people with unhealthy pulmonary sounds is 52.48, the standard deviation is 29.14 (Kababulut et al., 2021).

Table 3.1 Distribution of people from whom pulmonary sound recordings were obtained based on diagnosis

Healthy	**URTI**	**COPD**	**LRTI**	**Bronchiectasis**	**Pneumonia**	**Bronchiolitis**
26 (%20.64)	14 (%11.11)	64 (%50.79)	2 (%1.59)	7 (%5.55)	6 (%4.76)	6 (%4.76)

3.2 Our Datasets

As seen in Table 3.1, the records of patients with COPD constitute the majority of the dataset. This distribution makes it difficult for machine learning algorithms to learn classes other than COPD, which leads to a performance reduction. Therefore, we needed to make our dataset more balanced. First, we removed the records of LRTI patients due to insufficient number of data recordings. Then, we tried to make the dataset more balanced by data augmentation using two different methods.

3.2.1 Authentic Datasets

In order to make the distribution more homogeneous in this dataset, we augmented the data by taking the respiratory stages of the records of classes of less amount of data available as if they were separate recordings. In the annotation files in the database, there are informations about how many respiratory cycles a sound recording has and when respiratory cycles start and end. Using this information in the first method, we augmented the non-COPD records by the number of respiratory cycles. As shown in Table 3.2 to 3.6, we called these datasets as authentic datasets.

3.2.1.1 Two-Class Authentic Datasets

Dataset 1 is a dataset obtained by augmenting the non-COPD records by the number of respiratory cycles. Dataset 4 is a dataset in which MFCC features are extracted by dividing the records into 500 ms segments. Dataset 16 and all subsequent binary authentic datasets are datasets in which records with Linear Predictive Cepstral Coefficients (LPCC) values of 0 are deleted. In Dataset 16, the Unhealthy class includes the other 5 classes. In Dataset 24 and all subsequent binary authentic data sets, the Bronchectasy class is completely removed due to insufficient number of data. In Dataset 24, the Others class includes the URTI, Pneumonia and Bronchiolitis classes. In Dataset 25, the Others class includes the URTI and Bronchiolitis classes. In Dataset 27, the Others class includes the Pneumonia and Bronchiolitis classes. In Dataset 29, the Others class includes the URTI and Pneumonia classes. In Dataset 31,

the Others class includes 4 other classes except Bronchectasy. In the 3 datasets between Dataset 32 and 34 shown in Figure 3.1, a method was applied in which the decisions made as the Others (False Negative and True Negative) were carried to the next hierarchical classification. For example, in Dataset 34, the Others class decisions include the Healthy class as True Negative and the COPD, Pneumonia and Bronchiolitis classes as False Negative. In the three datasets between Dataset 37 and 39 depicted in Figure 3.1, a method was implemented where the decisions categorized as Others (only True Negative) were advanced to the subsequent hierarchical classification. In both of these methods, the number of classes decreases by one in the next classifications. For example, while there are 3 classes (Healthy, URTI, Pneumonia) in the Others class in Dataset 32 and 37, there are 2 classes (Healthy and URTI) in the next datasets, Dataset 33 and 38. However, in the datasets between 40 and 43 shown in Figure 3.1, there are always 4 classes in the Others class. These datasets do not have a hierarchical structure like the others. These datasets contain remaining all 5 classes after deleting data with LPCC values of 0 and the Bronchectasy class.

Table 3.2 Class distribution in two-class authentic datasets

Dataset	**Train**		**Test**	
Dataset 1	Healthy	Unhealthy	Healthy	Unhealthy
	214	1098	108	487
Dataset 4	Healthy	Unhealthy	Healthy	Unhealthy
	840	3006	386	1545
Dataset 16	Healthy	Unhealthy	Healthy	Unhealthy
	177	921	72	454
Dataset 24	COPD	The Others	COPD	The Others
	509	394	222	218
Dataset 25	Pneumonia	The Others	Pneumonia	The Others
	204	190	81	137
Dataset 26	URTI	Bronchiolitis	URTI	Bronchiolitis
	86	104	97	40
Dataset 27	URTI	The Others	URTI	The Others
	86	308	97	121

Table 3.2 Class distribution in two-class authentic datasets (continue)

Dataset 28	Pneumonia	Bronchiolitis	Pneumonia	Bronchiolitis
	204	104	81	40
Dataset 29	Bronchiolitis	The Others	Bronchiolitis	The Others
	104	290	40	178
Dataset 30	URTI	Pneumonia	URTI	Pneumonia
	86	204	97	81
Dataset 31	COPD	The Others	COPD	The Others
	509	571	222	290
Dataset 32	Bronchiolitis	The Others	Bronchiolitis	The Others
	104	467	40	252
Dataset 33	Pneumonia	The Others	Pneumonia	The Others
	204	263	81	190
Dataset 34	URTI	The Others	URTI	The Others
	86	177	92	115
Dataset 37	Bronchiolitis	The Others	Bronchiolitis	The Others
	104	467	40	250
Dataset 38	Pneumonia	The Others	Pneumonia	The Others
	204	263	81	164
Dataset 39	URTI	Healthy	URTI	Healthy
	86	177	92	72
Dataset 40	Bronchiolitis	The Others	Bronchiolitis	The Others
	104	976	40	472
Dataset 41	Pneumonia	The Others	Pneumonia	The Others
	204	876	81	431
Dataset 42	URTI	The Others	URTI	The Others
	86	994	97	415
Dataset 43	Healthy	Unhealthy	Healthy	Unhealthy
	177	903	72	440

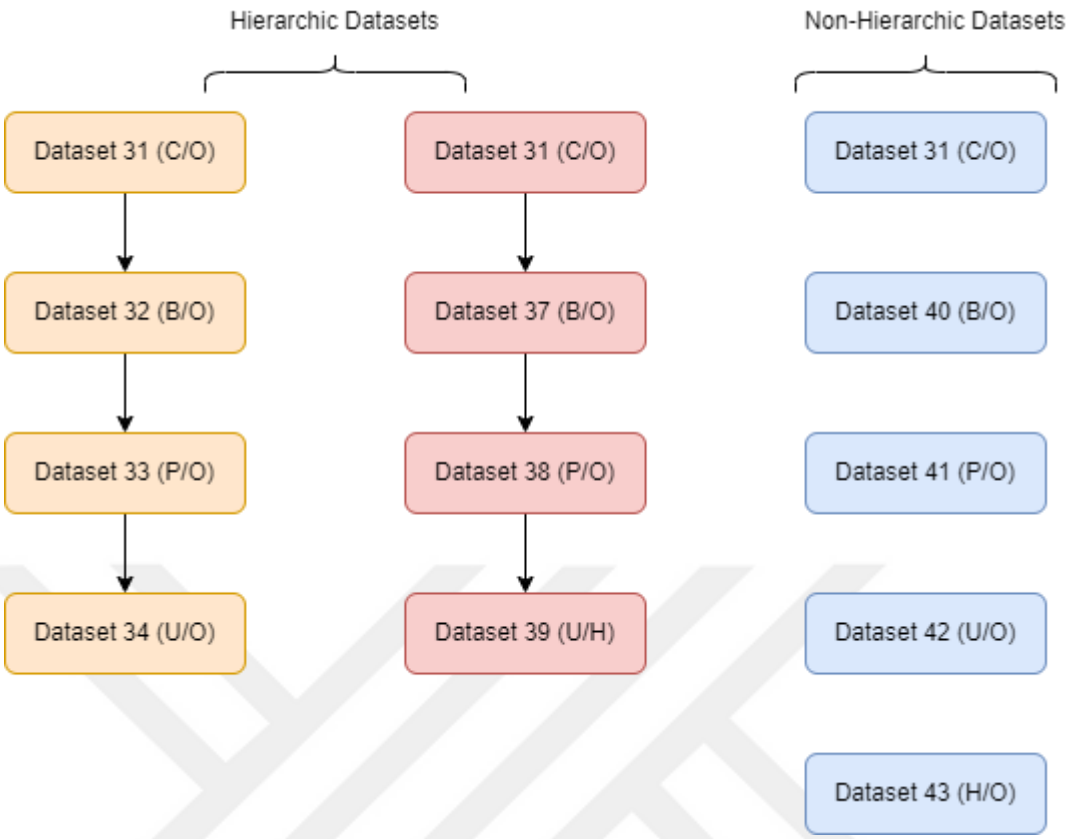


Figure 3.1 Hierarchic and non-hierarchic datasets

3.2.1.2 Three-Class Authentic Datasets

In Dataset 11, the Others class includes the Healthy, URTI, Pneumonia and Bronchiolitis classes. In Dataset 13, the Others class includes the URTI and Pneumonia classes. Dataset 17 and all subsequent three-class authentic datasets are datasets in which records with LPCC values of 0 are deleted. In Dataset 17, the Others class includes the Healthy, URTI, Pneumonia and Bronchiolitis classes. In Dataset 18, the Others class includes the URTI, Pneumonia and Bronchiolitis classes. Dataset 35 is a dataset where the classifications made as Others (False Negative and True Negative) in Dataset 33 are divided into URTI, Healthy and Others (COPD, Bronchiolitis, Pneumonia).

Table 3.3 Class distribution in three-class authentic datasets

Dataset	**Train**			**Test**		
Dataset 11	COPD	Bronchec.	The Others	COPD	Bronchec.	The Others
	555	73	684	238	31	326
Dataset 13	COPD	Bronchec.	The Others	COPD	Bronchec.	The Others
	555	73	470	238	31	218
Dataset 17	COPD	Bronchec.	The Others	COPD	Bronchec.	The Others
	509	18	571	222	14	290
Dataset 18	COPD	Bronchec.	The Others	COPD	Bronchec.	The Others
	509	18	394	222	14	218
Dataset 35	URTI	Healthy	The Others	URTI	Healthy	The Others
	86	177	66	92	72	43

3.2.1.3 Four-Class Authentic Datasets

Dataset 19 is a 4-class dataset in which records with LPCC values of 0 are deleted.

Table 3.4 Class distribution in four-class authentic datasets

Dataset	**Train**				**Test**			
Dataset 19	COPD	URTI	Pneu.	Bronchiol.	COPD	URTI	Pneu.	Bronchiol.
	509	86	204	104	222	97	81	40

3.2.1.4 Five-Class Authentic Datasets

Dataset 20 is a dataset containing 5 classes except COPD, where records with LPCC values of 0 are deleted. The difference between Dataset 36 and Dataset 20 is that it is a 5-class dataset in which the COPD class is included instead of the Bronchectasy class.

Table 3.5 Class distribution in five-class authentic datasets

Dataset	**Train**					**Test**				
Dataset 20	Heal.	URTI	Bronchec.	Pneu.	Bronchiol.	Heal.	URTI	Bronchec.	Pneu.	Bronchiol.
	177	86	18	204	104	72	97	14	81	40
Dataset 36	Heal.	URTI	COPD	Pneu.	Bronchiol.	Heal.	URTI	COPD	Pneu.	Bronchiol.
	177	86	509	204	104	72	97	222	81	40

3.2.1.5 Six-Class Authentic Datasets

Dataset 21 is a 6-class dataset in which records with LPCC values of 0 are deleted.

Table 3.6 Class distribution in six-class authentic datasets

Dataset	**Train**						**Test**					
Dataset 21	Heal.	URTI	COPD	Bronchec.	Pne.	Bronchiol.	Heal.	URTI	COPD	Bronchec.	Pne.	Bronchiol.
	177	86	509	18	204	104	72	97	222	14	81	40

3.2.2 Synthetic Datasets

With the second method, we obtained tabular synthetic data from MFCC, PSD and Statistical Features (STF) features using Generative Adversarial Network (GAN). Sometimes we added this data to the original data and sometimes we used only this synthetic data. As seen in Table 3.7 to 3.9, we called these datasets synthetic datasets. The numbers in parentheses in the tables indicate the numbers of synthetic data added to those classes.

3.2.2.1 Two-Class Synthetic Datasets

Dataset 2 is a dataset created by adding synthetic data to the Healthy class with the Adaptive Synthetic Sampling Approach (ADASYN) algorithm. Dataset 3 is a dataset where data is augmented by adding noise to Healthy class records. Dataset 5 is a data set created by adding synthetic data produced with GAN to the original MFCC features of the Healthy class and the original MFCC features of other non-COPD classes (URTI, Bronchectasy, Pneumonia, Bronchiolitis) in the Unhealthy class. Dataset 6 is a dataset created by adding synthetic data produced with GAN to the original MFCC features of the Healthy class and synthetically generating all MFCC features of the Unhealthy class with GAN. Dataset 7 is a dataset created by adding synthetic data produced with GAN to the original MFCC features of the Healthy class and synthetically generating all MFCC features of other non-COPD classes of the Unhealthy class with GAN. The distribution of Dataset 8 is the same as Dataset 6. The only difference of Dataset 8 is that its synthetic data is generated from the PSD feature. The difference of Dataset 9 from Dataset 5 is the number of MFCC synthetic data produced. The difference between Dataset 10 and Dataset 9 is that 2400 MFCC synthetic data are added to the Healthy class. Dataset 14 is a dataset created by synthetically generating all train MFCC features with GAN.

Table 3.7 Class distribution in two-class synthetic datasets

Dataset	**Train**		**Test**	
Dataset 2	Healthy	Unhealthy	Healthy	Unhealthy
	1098 (884)	1098	108	487
Dataset 3	Healthy	Unhealthy	Healthy	Unhealthy
	747 (533)	747	108	487
Dataset 5	Healthy	Unhealthy	Healthy	Unhealthy
	1215 (1001)	1215 (429)	108	487
Dataset 6	Healthy	Unhealthy	Healthy	Unhealthy
	1215 (1001)	1215 (1215)	108	487
Dataset 7	Healthy	Unhealthy	Healthy	Unhealthy
	1215 (1001)	1215 (972)	108	487

Table 3.7 Class distribution in two-class synthetic datasets (continue)

Dataset 8	Healthy	Unhealthy	Healthy	Unhealthy
	1215 (1001)	1215 (1215)	108	487
Dataset 9	Healthy	Unhealthy	Healthy	Unhealthy
	1414 (1200)	2598 (1500)	108	487
Dataset 10	Healthy	Unhealthy	Healthy	Unhealthy
	2614 (2400)	2598 (1500)	108	487
Dataset 14	Healthy	Unhealthy	Healthy	Unhealthy
	1200 (1200)	1500 (1500)	108	487

3.2.2.2 Three-Class Synthetic Datasets

Dataset 12 is a dataset created by adding synthetic data produced with GAN only to the original MFCC features of the Bronchectasy class. Dataset 15 is a dataset created by synthetically generating all train MFCC features with GAN. The distribution of Dataset 22 is the same as Dataset 15. The only difference of Dataset 22 is that its synthetic data is generated from the STF features.

Table 3.8 Class distribution in three-class synthetic datasets

Dataset	**Train**			**Test**		
Dataset 12	COPD	Bronchectasy	The Others	COPD	Bronchectasy	The Others
	555	673 (600)	684	238	31	326
Dataset 15	COPD	Bronchectasy	The Others	COPD	Bronchectasy	The Others
	1200 (1200)	1200 (1200)	1200 (1200)	238	31	326
Dataset 22	COPD	Bronchectasy	The Others	COPD	Bronchectasy	The Others
	1200	1200 (1200)	1200 (1200)	238	31	326

3.2.2.3 Six-Class Synthetic Datasets

Dataset 23 is a 6-class dataset created by adding synthetic data produced with GAN to all the original MFCC features in the train set.

Table 3.9 Class distribution in six-class synthetic datasets

Data set	**Train**						**Test**					
Data set 23	Heal.	URTI	COPD	Bronchec.	Pne.	Bronchiol.	Heal.	URTI	COPD	Bronchec.	Pne.	Bronchiol.
	600 (423)	600 (514)	600 (91)	600 (582)	600 (396)	600 (496)	72	97	222	14	81	40

CHAPTER FOUR
PULMONARY SOUND ANALYSIS METHODS

4.1 Features Used in Pulmonary Sound Analysis

Many machine learning algorithms require features to analyze and classify data. Although these algorithms use many different features in signal processing, when it comes to audio classification, the following three features dominate the field in terms of performance. These features are Mel Frequency Cepstral Coefficients (MFCC) (Aykanat et al., 2017; Bahoura & Pelletier, 2004; Bardou et al., 2018; Engin et al., 2022; Jaber et al., 2020; Naqvi & Choudhry, 2020; Palaniappan et al., 2014; Palaniappan & Sundaraj, 2013; Perna & Tagarelli, 2019; Rocha et al., 2021; Sengupta et al., 2016; Srivastava et al., 2021; Tabatabaei et al., 2021), Power Spectral Density (PSD) (Manir et al., 2020; Nilanon et al., 2016) and Statistical Features (Gavriely & Cugell, 2019; Naqvi & Choudhry, 2020; Priftis et al., 2018; Rao et al., 2019). In the following, we will examine these features in a little more detail.

4.1.1 Mel Frequency Cepstral Coefficients (MFCC)

Mel Frequency Cepstral Coefficients (MFCC) is one of the most commonly used features in sound analysis. The MFCC uses a method similar to the frequency selectivity of the human ear (Wang et al., 2016). While MFCC performs a linear detection at frequencies below 1000 Hz, it performs a logarithmic detection at frequencies above 1000 Hz (Dao et al., 2016). It consists of five stages. These are: Framing, Windowing, Discrete Fourier Transform (DFT), Mel Filter Bank and Discrete Cosine Transform (DCT).

4.1.1.1 Framing

Since the audio signal changes over time, it needs to be framed in small time intervals for analysis since in these short enough time frames, the sound can be assumed as approximately stationary (Singh & Rani, 2014). Audio data is passed

through these framing stage where frames overlaps to ensure smooth transitions (Çelebi, 2017). The number of samples for a frame is calculated as follows:

$$N = t.F_s \tag{4.1}$$

F_s is the sampling frequency, and *t* is the time interval of a frame.

4.1.1.2 Windowing

After the audio data is divided into frames, each frame is multiplied by a window function to highlight the desired frequencies. One well known window function is a rectangular window. However, discontinuities at the starting and ending points of this window cause problems in frequency analysis (Singh & Rani, 2014). Therefore, the Hamming window function, which prevents these discontinuities, is the most commonly used window in sound analysis (Singh & Rani, 2014). The process of windowing and the formula of the Hamming window function are shown as follows (Hamming, 1989).

$$\begin{aligned} Y(n) &= X(n)W(n) \\ W(n) &= 0,54 - 0,46\cos\left(\frac{2\pi n}{N-1}\right) \end{aligned} \tag{4.2}$$

W (n) is a Hamming window function defined between 0 and *N-1*.

4.1.1.3 Discrete Fourier Transform (DFT)

Discrete Fourier Transform (DFT) is the frequency representation of a discrete time input data. Digital systems use limited signals and these signals are discrete, because of this reason DFT is often used for frequency display. Therefore, we can see the frequency response of the output signal calculated in the previous section for each window with DFT as follows (Cooley et al., 1969).

$$Y(k)=\sum_{n=0}^{N-1}Y(n)e^{\frac{-j2\pi kn}{N}} \qquad k=0,1,2,3,\ldots,N-1 \tag{4.3}$$

4.1.1.4 Mel-Filter Bank

MFCC calculates the power of the all frequency bands of a signal by passing the signal through different band pass filters (Çelebi, 2017). Mel-filter bank contains a total of 40 overlapped triangle filters, 13 of linearly interval and 27 of logarithmic interval (Majidnezhad & Kheidorov, 2013). Mel-frequency conversion is done as shown below (Li & Chang, 2007):

$$\begin{aligned} B(f)&=1125\ln\left(1+\frac{f}{700}\right)\\ B^{-1}(b)&=700\left(10^{\frac{b}{1125}-1}\right)\end{aligned} \tag{4.4}$$

B(f) represents the scale of mel, and $B^{-1}(b)$ represents the frequency (Hz).

Mel scale center frequencies are converted to the centers of mel filters *f(m)* in Hertz using Equation 4.4 (Li & Chang, 2007). Mel filter banks for triangle filters are calculated as follows (Li & Chang, 2007).

$$M(m,k)=\begin{cases} 0\ldots\ldots\ldots\ldots\ldots\ldots k<f(m-1)\\ \dfrac{k-f(m-1)}{f(m)-f(m-1)}\ldots f(m-1)\le k\le f(m)\\ \dfrac{f(m+1)-k}{f(m+1)-f(m)}\ldots f(m)\le k\le f(m+1)\\ 0\ldots\ldots\ldots\ldots\ldots\ldots k>f(m+1)\end{cases} \tag{4.5}$$

An example of a mel filter bank is shown in Figure 4.1.

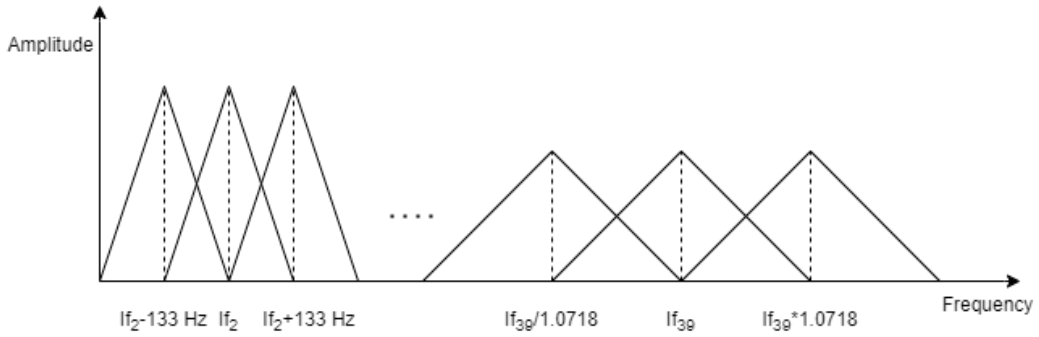


Figure 4.1 Mel-filter bank

Y(m) log energy are calculated by passing each window which were subjected to DFT, through the mel filter bank as shown below (Li & Chang, 2007).

$$Y(m) = \ln\left\{\sum_{k=0}^{N-1}|Y(k)|^2 M(m,k)\right\} \tag{4.6}$$

4.1.1.5 Discrete Cosine Transform (DCT)

Mel Frequency Cepstral Coefficients *c(n)* are calculated by Discrete Cosine Transform (DCT) as shown below (Li & Chang, 2007).

$$c(n) = \sum_{m=0}^{M-1} Y(m)\cos(\pi n(m-0,5)/M) \tag{4.7}$$

4.1.2 Power Spectral Density (PSD)

The Power Spectral Density (PSD) of a signal is equal to the Fourier Transform of the time-averaged autocorrelation of that signal as shown below (Stoica & Moses, 2005).

$$S_{xx}(w) = \lim_{T\to\infty} \frac{E\left\{|x_T(w)|^2\right\}}{2T} \tag{4.8}$$

If the signal is Wide Sense Stationary (WSS), the PSD is equal to the Fourier transform of the autocorrelation. This is called the Wiener-Khinchin Theorem (Champeney, 1987).

$$\text{If } R_{xx}(t, t+\tau) = R_{xx}(\tau) \text{ , then } S_{xx}(w) = \Im(R_{xx}(\tau)) \tag{4.9}$$

In this study, we computed the PSD using Welch's method. As a result of our calculations, we extracted 43 PSD features for each lung audio recording.

4.1.3 Statistical Features (STF)

In signal processing, statistical features are extracted from the statistical properties of a signal in the time domain. Statistical features are also used in lung sound classification. For example, in one study, statistical features such as mean and standard deviation are used to distinguish wheeze while kurtosis, a high-order statistical feature, is used for crackle (Gavriely & Cugell, 2019). The statistical features used are described in detail below.

1. *Mean:* It is the average of the amplitudes of all lung sound recordings.

2. *Standard Deviation*: It is the standard deviation of the amplitudes of all lung sound recordings.

3. *Skewness:* It is a measure of asymmetry of the probability distribution of the amplitudes of all lung sound recordings. Skewness is the standardized 3rd order moment and is calculated as follows.

$$\bar{\mu}_S = E\left[\left(\frac{x-\mu}{\sigma}\right)^3\right] \tag{4.10}$$

4. *Kurtosis:* It is a measure of curvature of the probability distribution graph of the amplitudes of all lung sound recordings. Kurtosis is the standardized order 4 moment and is calculated as follows.

$$Kurt[x] = E\left[\left(\frac{x-\mu}{\sigma}\right)^4\right] \tag{4.11}$$

5. *Root Mean Square:* It is the quadratic mean of the amplitudes of all lung sound recordings and is calculated as follows.

$$X_{RMS} = \sqrt{\frac{1}{n}\left(x_1^{\,2} + x_2^{\,2} + \ldots + x_n^{\,2}\right)} \tag{4.12}$$

6. *Mean Absolute Deviation*: It is the average of the absolute values of deviations to the mean of the amplitudes of all lung sound recordings and is calculated as follows.

$$D_{mean} = E\left|x-\mu\right| \tag{4.13}$$

7. *Median Absolute Deviation:* It is the median of the absolute values of deviations to the median of the amplitudes of all lung sound recordings and is calculated as follows.

$$D_{median} = Median\left|x - median(x)\right| \tag{4.14}$$

8. *Peak to Peak:* It is the difference between the maximum and minimum amplitudes of all lung sound recordings and is calculated as follows.

$$p2p = \max(x) - \min(x) \tag{4.15}$$

9. *Crest Factor:* It is the ratio of the peak value of the amplitudes of all lung sound recordings to the effective value and is calculated as follows.

$$X_{crest} = \frac{x_p}{\sqrt{\frac{1}{N}\sum_{i=1}^{N} x_i^{\,2}}} \tag{4.16}$$

10. *Shape Factor:* It is the ratio of RMS value of the amplitudes of all lung sound recordings to the the average of the absolute values and is calculated as follows.

$$X_{SF} = \frac{X_{RMS}}{\frac{1}{N}\sum_{i=1}^{N} |x_i|} \tag{4.17}$$

11. *Impulse Factor:* It is the ratio of the peak value of the amplitudes of all lung sound recordings to the average of the absolute values and is calculated as follows.

$$X_{IF} = \frac{x_p}{\frac{1}{N}\sum_{i=1}^{N} |x_i|} \tag{4.18}$$

4.2 Machine Learning Algorithms Used in Pulmonary Sound Analysis

Machine learning algorithms are generally used in image processing in the biomedical field, but they are also used to analyze one-dimensional data such as sound. In this study, we used 7 different machine learning algorithms that are most preferred for the diagnosis of lung diseases from lung sounds.

4.2.1 Decision Tree

Decision trees are a supervised learning algorithm mostly used in classification problems. In the simplest decision trees, there is a structure with decisions in the leaves of the tree, which is formed by dividing into different branches as a result of the questions asked. In decision trees in complex classification problems, there are features instead of questions and classes instead of decisions. The top node of the tree

starts with the feature with the highest information gain, and branching is shaped according to the information gain values of the other features (Kababulut et al., 2023). In order to find the information gain, Gini Impurity values of the features must first be calculated as in Equation 4.19 (Kababulut et al., 2023).

$$GI_k = 1 - \sum_{i=1}^{k} p_i(T, x)^2 \tag{4.19}$$

$p_i(T, x)$ is the probability of the feature of data x in class i with respect to the test data T (Kababulut et al., 2023). The Conditional Gini Impurity of the feature of data x with respect to the test data T is calculated as follows (Kababulut et al., 2023).

$$CG(T, x) = \sum_{k=1}^{l} p_k GI_k \tag{4.20}$$

p_k is the probability of the feature of the data in class k and GI is the Gini Impurity value of this feature (Kababulut et al., 2023). To determine the feature of the leading node, the information gain is calculated as in the equation below (Kababulut et al., 2023).

$$IG(T, x) = GI(T) - CG(T, x) \tag{4.21}$$

Once the feature at the first node is calculated, the other features are selected using the same logic and the decision tree is built (Kababulut et al., 2023).

4.2.2 Random Forest

Random Forest algorithm, as the name suggests, is a kind of machine learning algorithm that consists of more than one decision tree and is used in classification and regression problems. In this algorithm, trees are randomly divided into branches until they reach the decision stage. For example, in a classification problem, trees always give one vote for any class, and the random forest decides the class with the most votes

(Breiman, 2001). As the number of trees increases, the generalization error converges to a certain value (Breiman, 2001). This value is calculated as follows.

$$GE = P_{A,B}\left[P_r(Cl(A,r) = B) - \max_{C \neq B} P_r(Cl(A,r) = C) < 0\right] \quad (4.22)$$

GE is the generalization error, *A* is the input variable, *B* is the vote belong to correct class for the input variable, *r* is the random variable, *Cl* is a classifier, and *C* is the vote belongs to other classes for the input variable. An upper limit for generalization error can be determined in terms of power, which is a measure of how accurately a set of classifiers decide, and correlation, which is a measure of the dependence between them (Breiman, 2001). For this, we first need to define the margin function as seen in Equation 4.23 (Breiman, 2001).

$$m(A,B) = P_r(Cl(A,r) = B) - \max_{C \neq B} P_r(Cl(A,r) = C) \quad (4.23)$$

We can say that the margin function is the difference between the probability of the classifier choosing the right class and the probability of choosing other classes (Breiman, 2001). *s*, the power of the classifier ensemble, is calculated as follows (Breiman, 2001).

$$s = E_{A,B} m(A,B) \quad (4.24)$$

Raw margin function is calculated as follows (Breiman, 2001).

$$rm(r,A,B) = I(Cl(A,r) = B) - I(Cl(A,r) = \overline{C}(A,B)) \quad (4.25)$$

I(.) represents an indicator function (Breiman, 2001). The margin function is actually equal to the mean of the raw margin function with respect to the variable *r* (Breiman, 2001). Keeping the *r* and *r'* variables constant, the average of the correlation between the raw margins is calculated as follows (Breiman, 2001).

$$\overline{cor} = E_{r,r'}(cor(r,r')sd(r)sd(r')) / E_{r,r'}(sd(r)sd(r')) \quad (4.26)$$

sd is the standard deviation of the *rm*(*r*,*A*,*B*) function when the variable *r* is kept constant (Breiman, 2001). Finally, we can calculate the upper band of the Generalization Error as follows (Breiman, 2001).

$$GE \leq \overline{cor}(1-s^2) / s^2 \quad (4.27)$$

4.2.3 XGBoost Algorithm

XGBoost algorithm is also decision tree-based and has a similar approach with the gradient boosting algorithm, but has a significant speed and performance advantage over this algorithm. Gradient boosting algorithm is an algorithm that turns weak classifiers (Decision Trees) into strong classifiers by reducing estimation errors with gradient descent method. XGBoost, on the other hand, uses the Newton-Raphson method instead of gradient descent (Putrada et al., 2022) and performs pruning, which means removing unimportant branches from the tree. In XGBoost, the weights on the leaves are calculated as follows (T. Chen & Guestrin, 2016).

$$w_k = -\frac{\sum_{j \in I_k} g_j}{\sum_{j \in I_k} h_j + \lambda} \quad (4.28)$$

I_k is the sample set for leaf *k*, g_j and h_j are the first and second order derivatives of the prediction error over the prediction, and λ is a regulation parameter that prevents overfitting (T. Chen & Guestrin, 2016). The loss function, in which the split branches of a tree are evaluated, is calculated as follows (T. Chen & Guestrin, 2016).

$$Loss = -\frac{1}{2}\sum_{k=1}^{T}\frac{(\sum_{j\in I_k} g_j)^2}{\sum_{j\in I_k} h_j + \lambda} + \gamma T \tag{4.29}$$

γ is a parameter that determines the amount of pruning and T is the number of leaves (T. Chen & Guestrin, 2016).

4.2.4 K-Nearest Neighbour (K-NN)

The K-nearest neighbor (K-NN) algorithm is also a supervised machine learning algorithm used in classification problems. In K-NN, the class of a test data is determined by looking at the class of the data closest to it. For this, the following steps are applied:

1- All data, including test data, are positioned in n-dimensional space (Kababulut et al., 2021).

2- The k (k = positive integer) closest data to the test data are determined by calculations based on distance metrics (Kababulut et al., 2021). Although there are many distance metrics in the literature, we used the Euclidean distance in this study.

3- Whichever class is in the majority among the k closest data, the test data is labeled with that class (Kababulut et al., 2021).

We will examine the distance calculations used to determine the nearest K sample;

Manhattan Distance: This distance is calculated by the sum of the absolute differences of the coordinates of the two samples. Here, x_i and y_i are two samples with cartesian coordinates (Chomboon et al., 2015). The distance between x_i and y_i is calculated as in Equation 4.30 (Chomboon et al., 2015).

$$Man.Dist. = \sum_{i=1}^{n} |x_i - y_i| \tag{4.30}$$

Euclidean Distance: This distance is calculated by the square root of the sum of the squares of the differences of the coordinates of the two samples. The distance between x_i and y_i is calculated as in Equation 4.31 (Chomboon et al., 2015).

$$Euc.Dist. = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \tag{4.31}$$

Minkowski Distance: This distance calculation is essentially the generalization of Manhattan and Euclidean distances (Çelebi, 2017). The distance between x_i and y_i is calculated as in Equation 4.32 (Chomboon et al., 2015).

$$Min.Dist. = (\sum_{i=1}^{n} |x_i - y_i|^m)^{1/m} \tag{4.32}$$

Chebishev Distance: This distance is calculated by the maximum of the absolute differences of the coordinates of the samples. The distance between x_i and y_i is calculated as in Equation 4.33 (Chomboon et al., 2015).

$$Cheb.Dist. = Max_{i=1}^{n} |x_i - y_i| \tag{4.33}$$

Mahalanobis Distance: This distance is calculated by the covariance matrix of points. The distance between x_i and y_i is calculated as in Equation 4.34 (Chomboon et al., 2015).

$$Mah.Dist. = \sqrt{(X - Y)' C' (X - Y)} \tag{4.34}$$

The performance of the K-NN algorithm varies according to the distance calculation and the k value. Euclidean distance is the most commonly used calculation. The choice of k depends on the dataset. If the noise is high in the data, the k value is selected as large, and if we want more clarity between classes, the k value is selected as small (Çelebi, 2017).

4.2.5 Support Vector Machine (SVM)

Support Vector Machine (SVM) is also one of the supervised machine learning algorithms. In SVM, a non-classed sample is classified by looking at the position relative to the plane that separates the classes of certain samples (labeled). This classification consists of 3 stages;

1- All samples, including the new sample, are placed in n-dimensional space according to their coordinates (Hearst et al., 1998).

2- There are many planes separate the classes according to the placement of the samples (Çelebi, 2017). However, there is one separator plane that maximizes the margin, the distance between the closest points of each class (Hearst et al., 1998). This separator plane is determined by a number of analytical calculations.

3- The new sample is assigned to the class in which section of the determined separator plane it remains (Çelebi, 2017).

If the data set can be separated linearly, linear support vector machines are used, otherwise, non-linear support vector machines are used.

Linear Support Vector Machines: We suppose the data set contains two classes. Since these data have *n* features, the samples become an n-dimensional x_i vector ($x_i \in R^n$) and these samples belong to one of two classes $y_i \in (-1,1)$ (Hearst et al., 1998). The *x* points on the separator plane must satisfy Equation 4.35 (Hearst et al., 1998).

$$f(x) = sign(w' x_i + b) \tag{4.35}$$

In Equation 4.35, *w* denotes the normal of the separator plane and *b* is defined as the plane constant (Hearst et al., 1998). These support vectors are on parallel planes equidistant to the separator plane, which also form the boundary planes of the classes (Çelebi, 2017). The linear SVM tries to find the separator plane that maximizes the margin, the distance between these boundary planes (Hearst et al., 1998). We assume that all samples satisfy the inequality in Equation 4.36 (Hearst et al., 1998).

$$y_i \times (w' x_i + b) \geq 1 \tag{4.36}$$

The distance between the boundary planes is calculated as $\frac{2}{\|w\|}$. The minimum value $\frac{2}{\|w\|}$ for the separator plane that maximizes the margin, must satisfy Equation 4.36 (Hearst et al., 1998). We can find the separator plane as in Equation 4.37 under the conditon of $w = \sum_{i}^{n} v_i x_i$ (Hearst et al., 1998).

$$g(x) = \sum_{i}^{n} \left(\alpha_i z_i x_i\right)^{T} x_j + w_0 \tag{4.37}$$

As in Equation 4.37, it is seen that the separator plane is equal to the inner product of the feature vector (Çelebi, 2017).

Nonlinear SVM: When the data set cannot be separated by a linear separator plane, the nonlinear SVM realizes the separation by assigning the samples to a higher dimensional space (Çelebi, 2017). We suppose that we move the samples to another space with the ϕ assignment function (Çelebi, 2017). Since we will try to find the

separator plane in this space, the inner product in the equation of the linear separator plane is as in Equation 4.38 (Hearst et al., 1998).

$$K(x, y) = \phi(x)\phi(y) \tag{4.38}$$

In Equation 4.38, *y* represents the support vectors and *K* represents the kernel function (Çelebi, 2017). If the kernel function is written in Equation 4.37, the separator plane equation in this space can be found as in Equation 4.39 (Hearst et al., 1998).

$$f(x) = sign(\sum_{i=1}^{n} v_i K(x, x_i) + b) \tag{4.39}$$

We will examine the most commonly used Kernel functions (Hearst et al., 1998).

Polynomial Kernel Function: $K(x, y) = \langle x.y \rangle^m$ (4.40)

In Equation 4.40, *m* represents dimensions of R^N (Hearst et al., 1998).

Sigmoid Kernel Function: $K(x, y) = \tanh\left(G\langle x.x_i \rangle + \Theta\right)$ (4.41)

In Equation 4.41, *G* represents gain and Θ represents offset (Hearst et al., 1998).

Radial Basic Kernel Function: $K(x, x_i) = \exp^{-\frac{|x.x_i|^2}{2\sigma^2}}$ (4.42)

In Equation 4.42, σ is a parameter (Schölkopf et al., 2004).

4.2.6 Artificial Neural Network (ANN)

Artificial neural networks are software designed to work like a nerve cell, influenced by the learning of the human brain. Neural networks consist of neurons, an

input layer and an output layer. Multilayer perceptron neural networks additionally have one or more hidden layers between the input and output layers. The input data to the neurons in the input layer are multiplied by certain weights and summed one by one. The output of certain neuron is calculated as in Equation 4.43.

$$y = \sum_{j=1}^{n} w_j x_j + w_0 \tag{4.43}$$

w_j are the weights of the neuron and x_j are the input data of the neuron. These sum values are then passed through an activation function to determine the output values of the neurons. The output values of the neurons in the input layer are the inputs of the neurons in the hidden layer. In the same logic, the outputs of the neurons in the hidden layer are connected to the neurons in the output layer, and their outputs are the final decisions of the system. Activation functions are linear or non-linear functions such as Hard Limit, Linear, Sigmoid, Hyperbolic Tangent and Rectifier Linear Unit (ReLU). We used the ReLU function in our neural network. In the ReLU function, negative values become 0 while positive values remain the same. Learning in artificial neural networks is done by back-propagation algorithm. In the training phase, the network produces outputs with random weights given to the neurons. We call the difference(s) between these output/s and the actual output/s as error. With this algorithm, these errors are tried to be reduced by changing the weights given to the neurons at each epoch in the training phase of the network. The training phase is over when the error(s) reach(es) the minimum value(s), that is, when they/it no longer decrease(s) further. One of these error reduction and minimum error finding methods is the Stochastic Gradient Descent (SGD) method. The difference of the Stochastic Gradient Descent method from the Gradient Descent method is that the derived error function is updated for the outputs of randomly selected neurons, not for the entire training set (Samadianfard et al., 2022).

4.2.7 Logistic Regression

Regression is an algorithm that tries to find the relationship between two or more variables mathematically. Logistic regression tries to model this relationship with a sigmoic function (Messaoudi et al., 2022).

$$\overline{Y} = \frac{1}{1+e^{-F(x)}} \tag{4.44}$$

$\overline{Y}$ is the prediction of the dependent variable *Y*, *x* is the independent variable and *F* is a function that shows the mathematical relationship between them. Logistic Regression, which tries to find the correlation function by minimizing the error between the predicted value and the target value, uses regularization methods against overfitting, which is frequently encountered in classification problems (Messaoudi et al., 2022). The most commonly used regularization methods are L1 and L2 regularization and Elastic Net (Messaoudi et al., 2022) which combines these two methods. In regulation methods, the cost function to be minimized consists of loss and regulation terms. While the loss terms are the same in L1 and L2 regulation, there is a slight difference in the regulation term. L1 regularization eliminates unimportant features by taking the absolute values of the weight values (Messaoudi et al., 2022). L2 regularization takes the squared values of the weight values, resulting in a more accurate modeling (Messaoudi et al., 2022).

CHAPTER FIVE
VISUALIZATION OF DATA DISTRIBUTION

5.1 The Self Organized Map (SOM) Networks

The Self Organized Map (SOM) algorithm is an artificial neural network that makes a two-dimensional visualization of a dataset. In this network, the initial weights of the neurons are determined randomly. These weights are compared with an input data selected from the dataset, and the neuron with the most similar weight (the winning neuron) is selected. In addition, the weights of neurons adjacent to the winning neuron are updated.

Weight vectors paired with neurons are called codebook or reference vectors (Bieroza et al., 2012). The reference vectors are refreshed with each new input data, and the two-dimensional visual map of this dataset is completed when the training is finished.

5.1.1 U-Matrix Maps

The U-matrix map is the map, where the distances between neurons in the SOM network are seen. Figure 5.1 shows the U-Matrix map of the SOM network designed for the iris flower classification problem (Mathworks, 2023). In Figure 5.1, neurons are represented by blue hexagons in the U-matrix map. Neighboring neurons are connected with each other by red lines. Neighboring neurons are shown in light color if the distances are small, and in dark colors if they are large. Neurons surrounded by light colors can be thought of as a cluster of this dataset, and dark colors can be thought of as the boundaries separating these clusters. By using U-matrix maps, the number of clusters in the datasets and the distances of these clusters to each other can be determined. When we look at Figure 5.1, it is understood that the dark red border extending from the upper left part of the U-Matrix map to the lower left part divides the data into two classes (Mathworks, 2023).

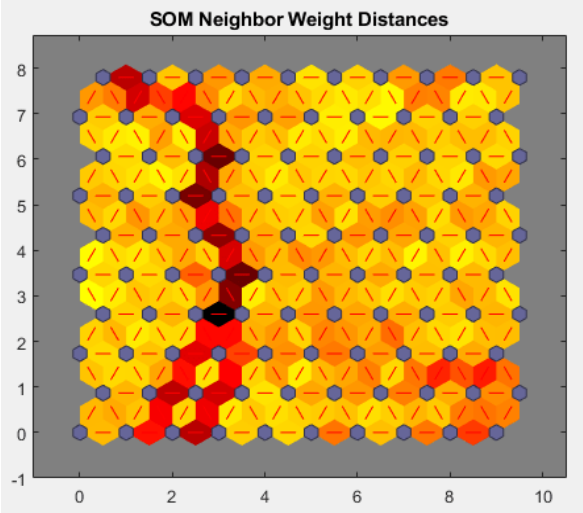


Figure 5.1 U-matrix map of iris flower dataset (Mathworks, 2023)

5.1.2 Somhit Maps

Somhit maps are the visuals that allow us to see which input data hit which neurons in the SOM network, that is, which neurons are the winners. Figure 5.2 shows the SOM-Hit map of the SOM network designed for the iris flower classification problem (Mathworks, 2023). When we look at Figure 5.2, we can see how many winning neurons were hit on which data.

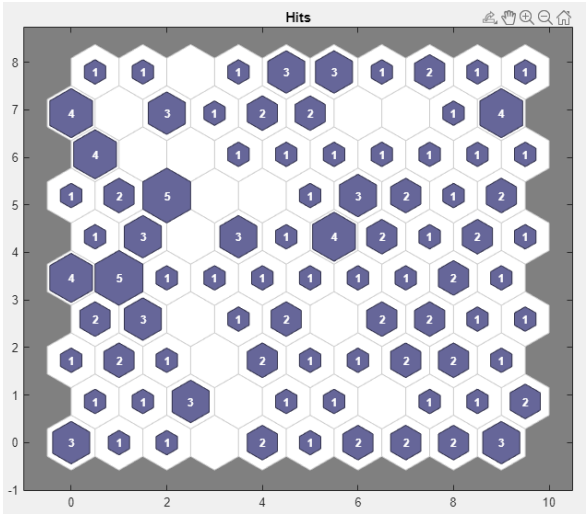


Figure 5.2 Somhit map of iris flower dataset (Mathworks, 2023)

5.1.3 Interpretation of SOM Network Maps

When we look at the Somhit maps of all three features, we can see that the Healthy (1st Class) class and URTI (2nd Class), Pneumonia (5th Class) and Bronchiolitis (6th Class) classes are intertwined. If we talk about the MFCC feature, we can say that there is some interference between the Healthy class and the Bronchectasy (4th Class) class. Because it can be seen from the Somhit map that most of the samples in the Bronchectasy class hit different neurons from the Healthy class. We can say from both the U-matrix and Somhit maps that the Healthy class and the COPD (3th Class) class are almost completely separable for the MFCC feature. This class distinction can be clearly seen from the dark cells going from the upper middle to the lower right part of the U-matrix map (there are neurons in this region that completely hit the COPD samples in the Somhit map). The Bronchectasy samples for the PSD feature appear to hit the same neurons as the Healthy samples. Although there is not a clear distinction for COPD and Healthy classes as in MFCC, it is seen that COPD samples hit different neurons from the Healthy class in the upper right side of the Somhit map. It is clearly seen that the COPD samples in the upper right are far from the other class samples in the U-matrix map. For statistical features, although it is seen that some of the Bronchectasy samples hit different neurons from the Healthy samples, a clear distinction cannot be made. Although there is not a clear distinction for COPD and Healthy classes as in MFCC, it is seen that COPD samples hit fewer different neurons in the lower right part of the map compared to other features. It is clearly seen that the few COPD samples in this lower right part are far from other class samples in the U-matrix map. After these U-matrix and Somhit maps, we can say that the reason why the algorithms cannot clearly distinguish the classes except for the COPD class is the confusion of the classes.

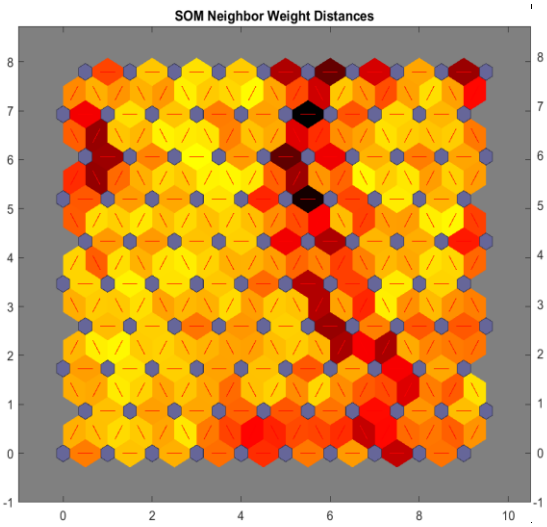


Figure 5.3 U-matrix of all MFCC features

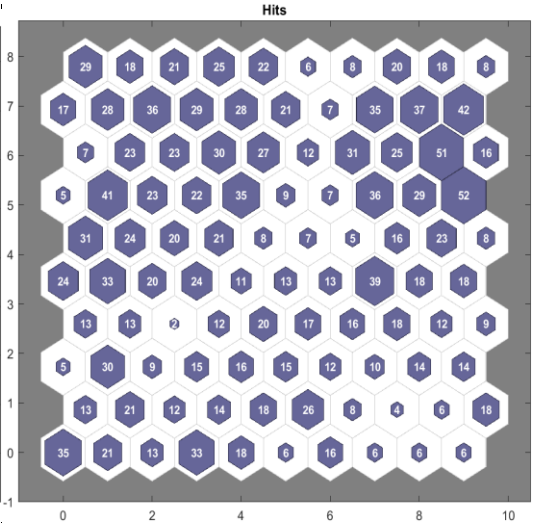


Figure 5.4 Somhit of all MFCC features

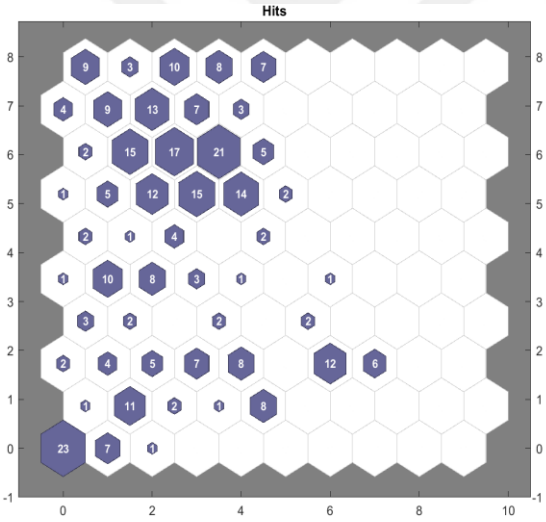


Figure 5.5 Somhit of healthy MFCC features

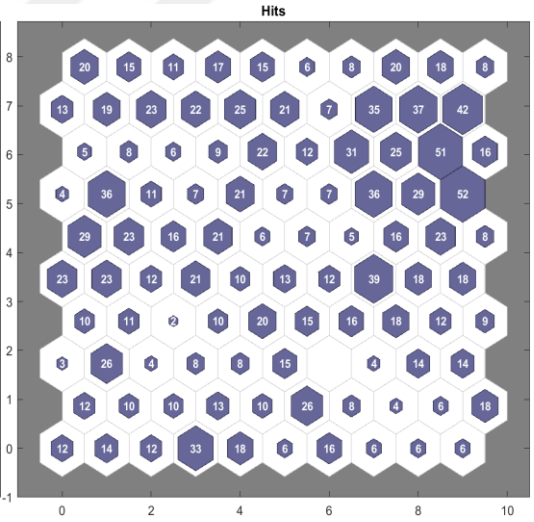


Figure 5.6 Somhit of unhealthy MFCC features

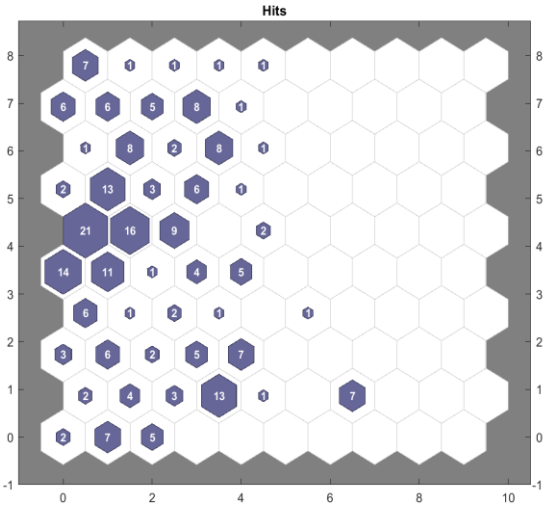


Figure 5.7 Somhit of URTI MFCC features

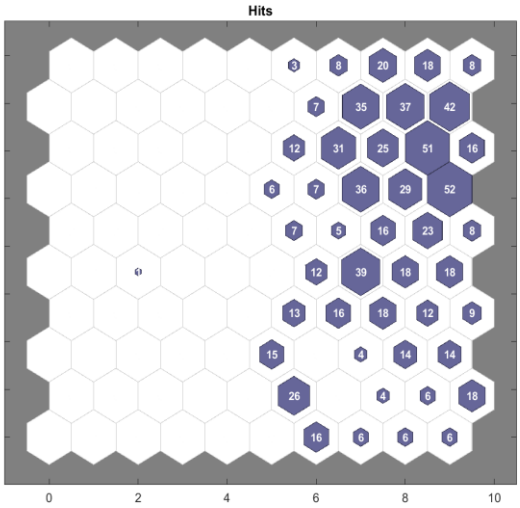


Figure 5.8 Somhit of COPD MFCC features

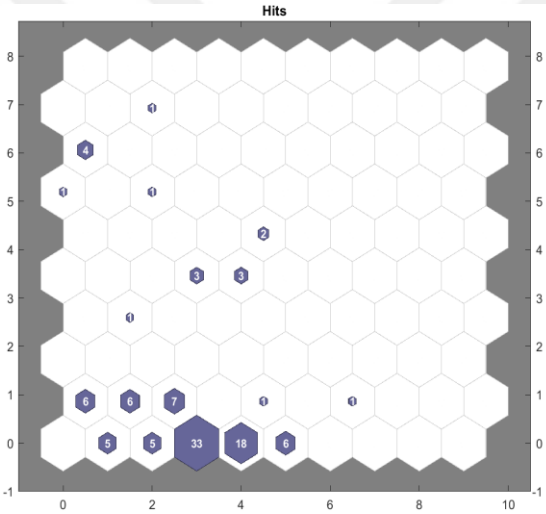


Figure 5.9 Somhit of Bronchiectasy MFCC feat.

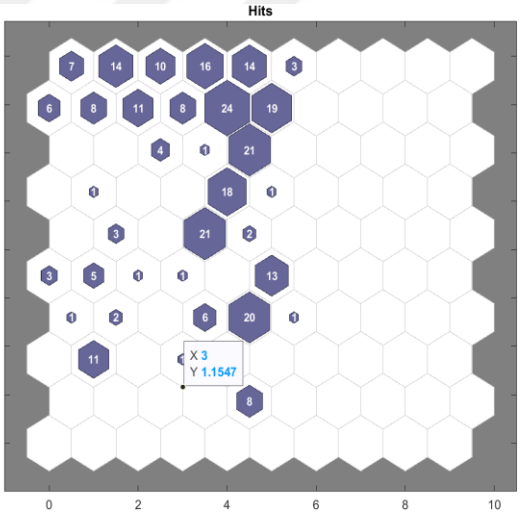


Figure 5.10 Somhit of Pneumonia MFCC feat.

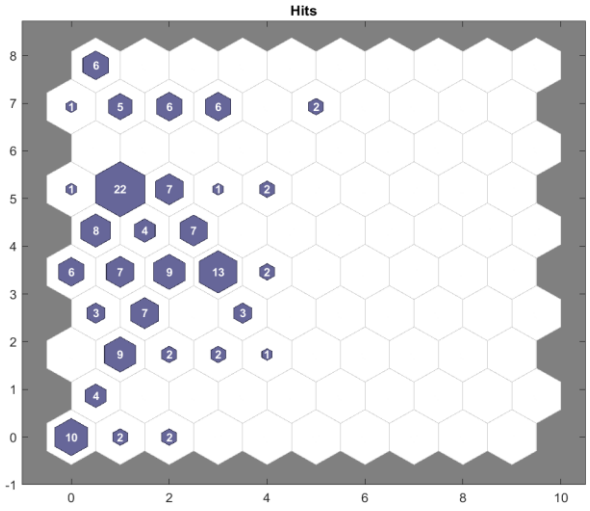


Figure 5.11 Somhit of Bronchiolitis MFCC features

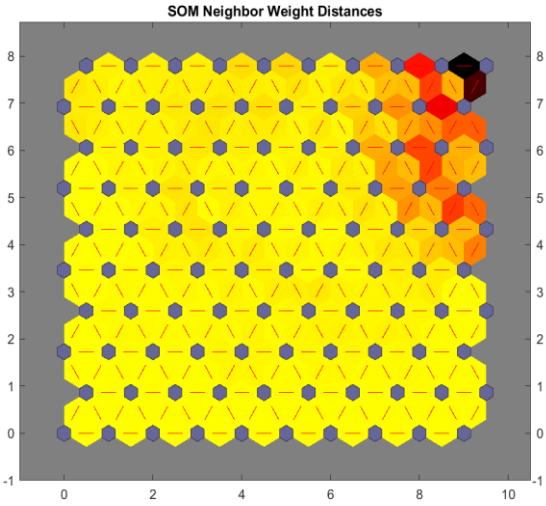


Figure 5.12 U-matrix of all PSD features

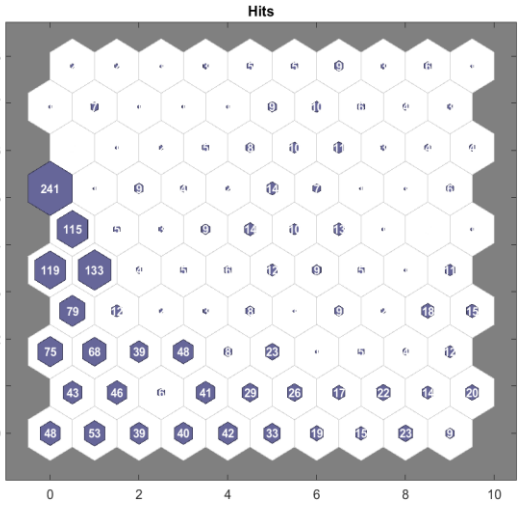


Figure 5.13 Somhit of all PSD features

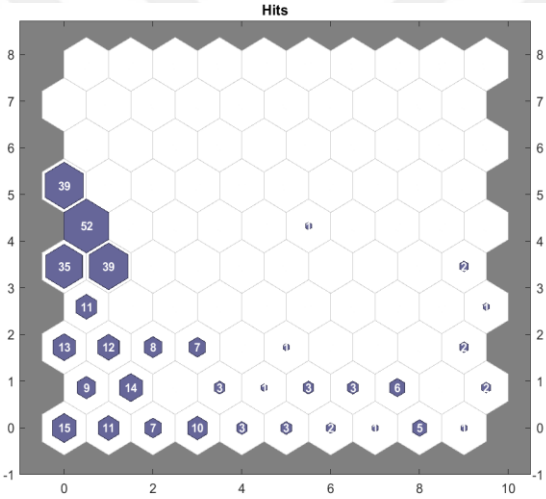


Figure 5.14 Somhit of healthy PSD features

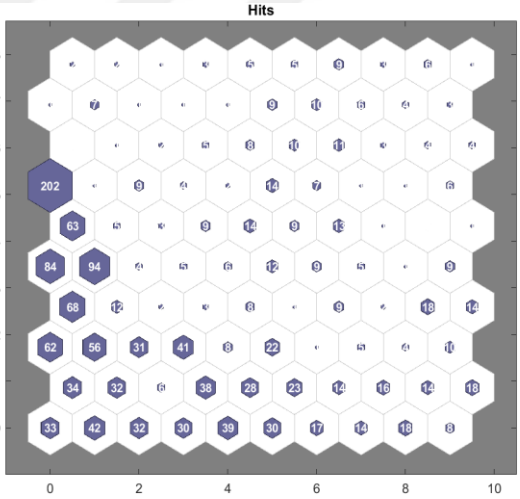


Figure 5.15 Somhit of unhealthy PSD features

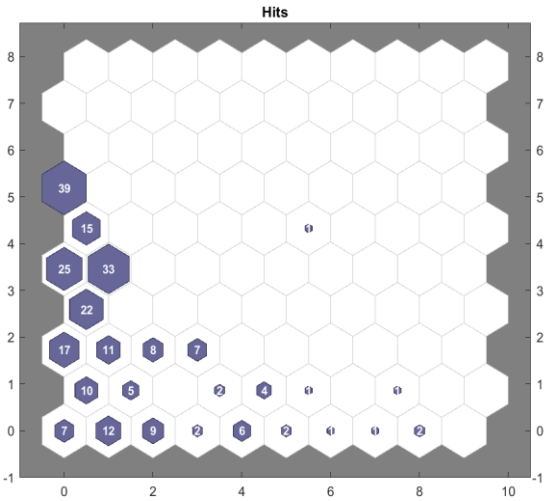


Figure 5.16 Somhit of URTI PSD features

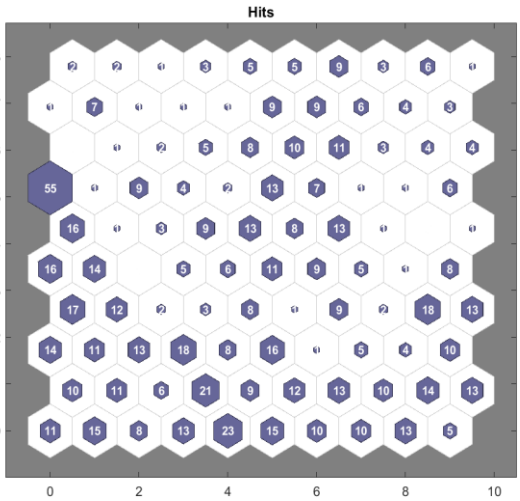


Figure 5.17 Somhit of COPD PSD features

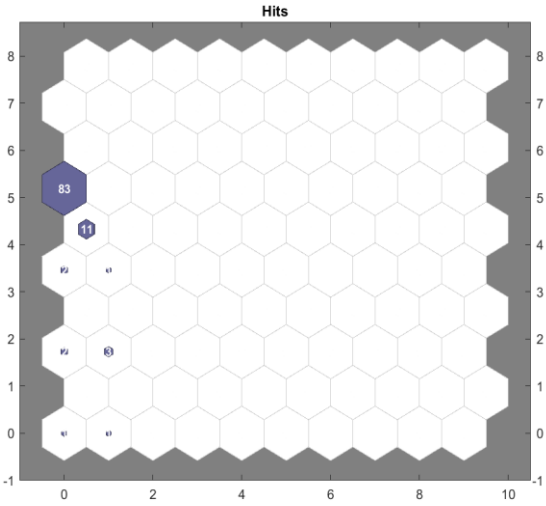


Figure 5.18 Somhit of Bronchiectasy PSD feat.

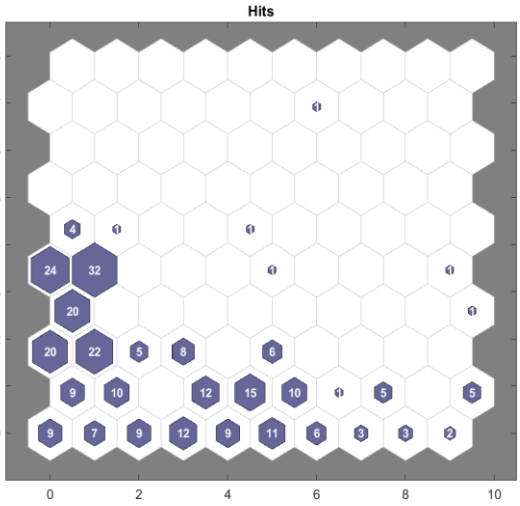


Figure 5.19 Somhit of Pneumonia PSD feat.

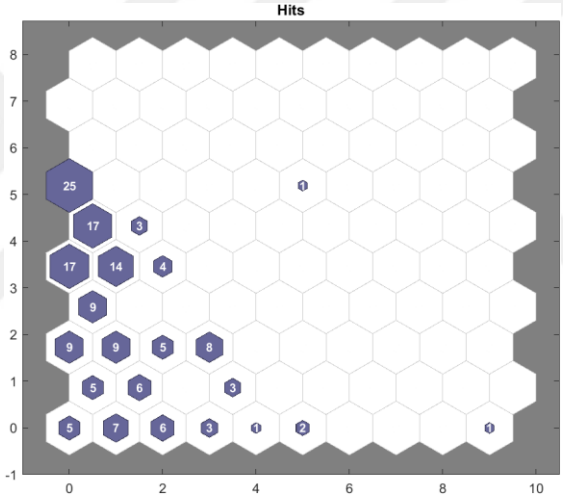


Figure 5.20 Somhit of Bronchiolitis PSD features

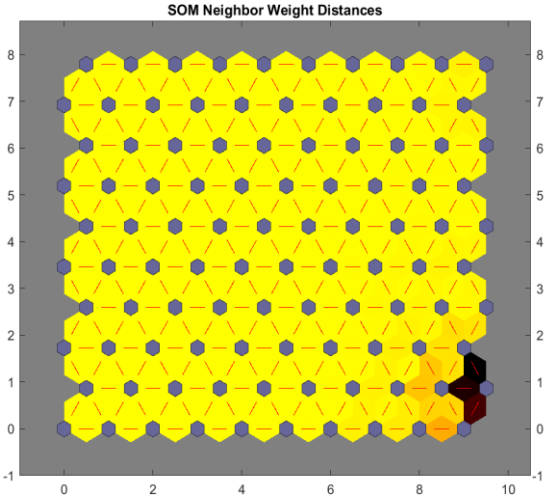


Figure 5.21 U-matrix of all STF features

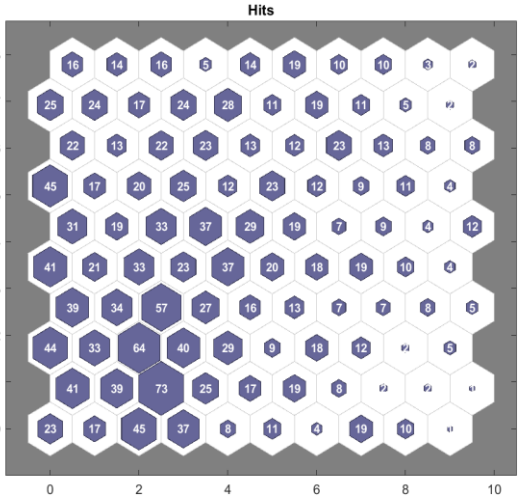


Figure 5.22 Somhit of all STF features

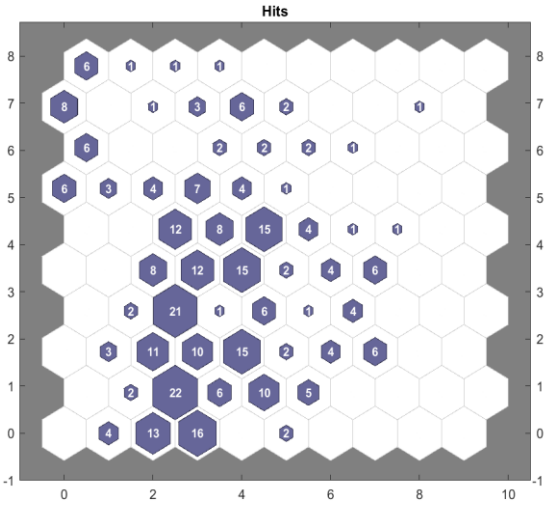


Figure 5.23 Somhit of healthy STF features

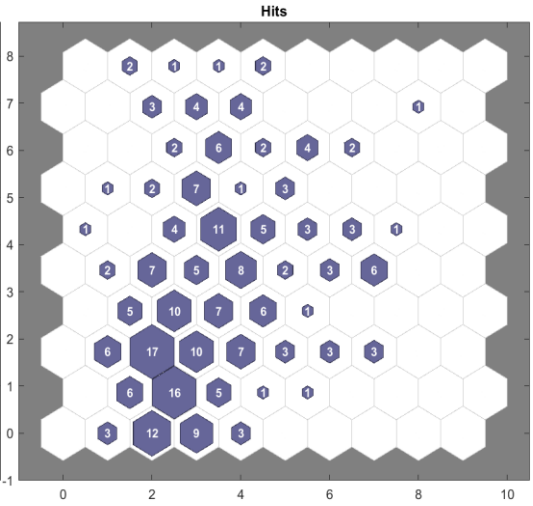


Figure 5.24 Somhit of URTI STF features

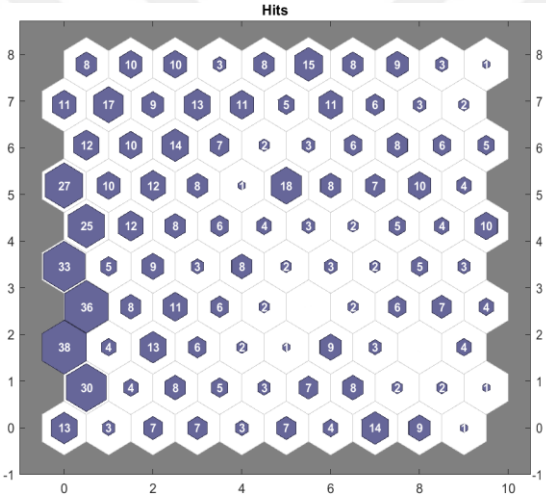


Figure 5.25 Somhit of COPD STF features

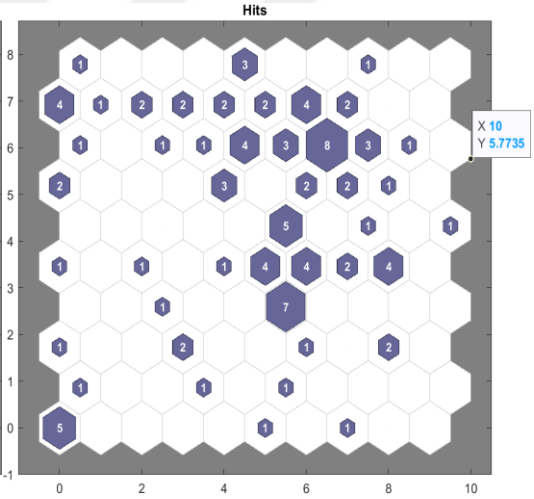


Figure 5.26 Somhit of Bronchiectasy STF feat.

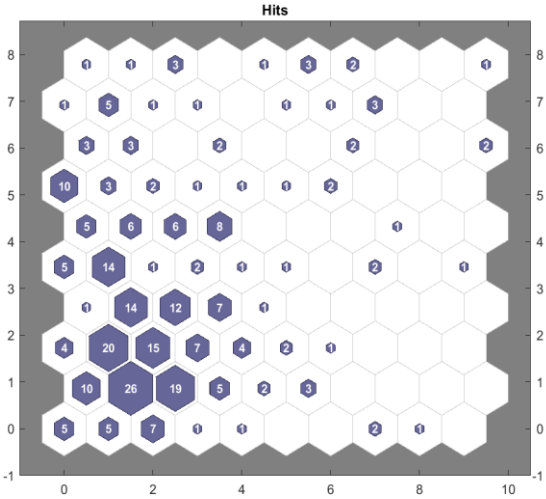


Figure 5.27 Somhit of Pneumonia STF feat.

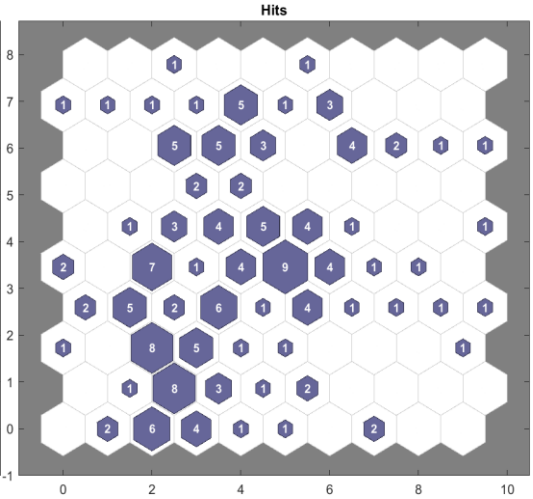


Figure 5.28 Somhit of Bronchiolitis STF feat.

5.2 The t-Distributed Stochastic Neighbor Embedding (t-SNE) Method

The t-Distributed Stochastic Neighbor Embedding (t-SNE) method is a statistics-based method that visualizes the class structure of multidimensional datasets in two and three dimensions. In this method, the probability distribution of the dataset that is desired to be visualized is obtained by assigning a higher probability value to the samples close to each other according to the Euclidean distance. A two- or three-dimensional model is then created, similar to the probability distribution of this dataset. By minimizing the Kullbak-Leibler deviation between these two distributions, the accuracy of the model is maximized (Van Der Maaten & Hinton, 2008). Thus, a visual map of the class distribution of this dataset emerges.

5.2.1 t-SNE Outputs of 6 Class-Distribution

1 represents “Healthy” class (Red Dots), 2 represents “URTI” class (Yellow Dots), 3 represents “COPD” class (Green Dots), 4 represents “Bronchiectasis” class (Light Blue Dots), 5 represents “Pneumonia” class (Dark Blue Dots) and 6 represents “Bronchiolitis” class (Purple Dots) in Figures 5.29-5.31 t-SNE outputs. These figures were extracted separately for 3 different features (MFCC, PSD, STF).

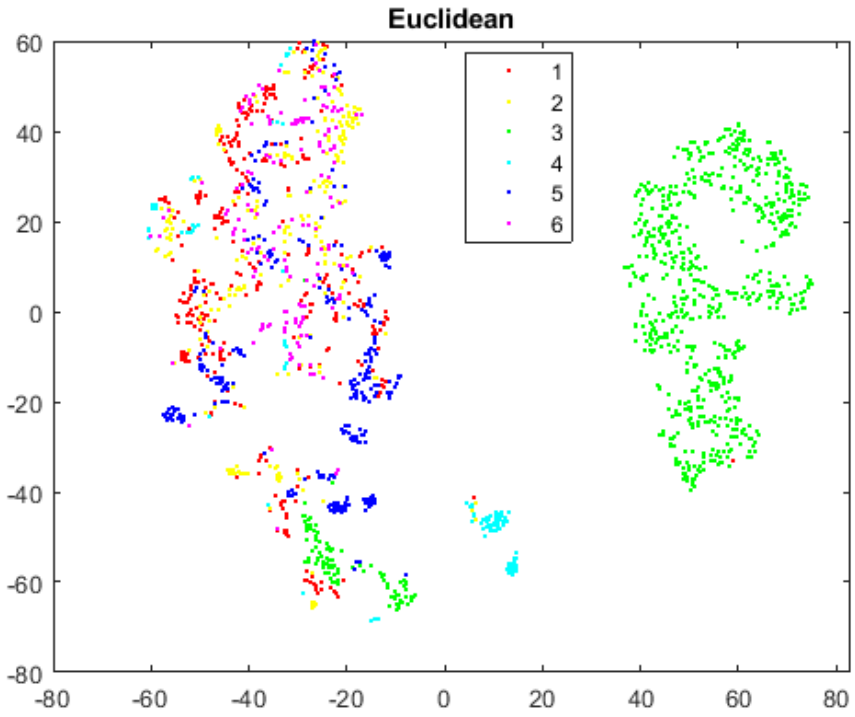


Figure 5.29 t-SNE of 6 class MFCC features

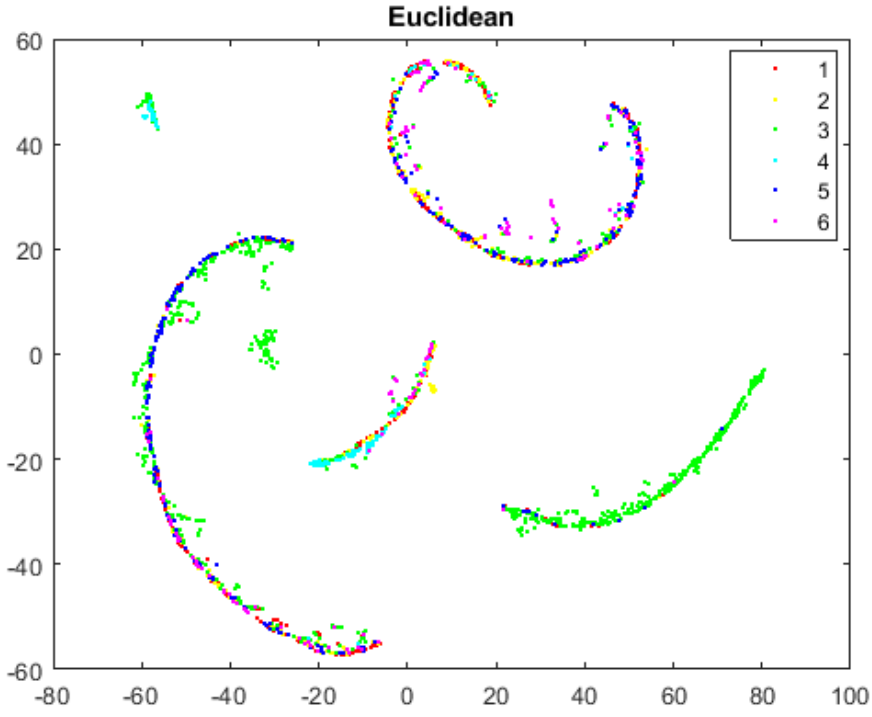


Figure 5.30 t-SNE of 6 class PSD features

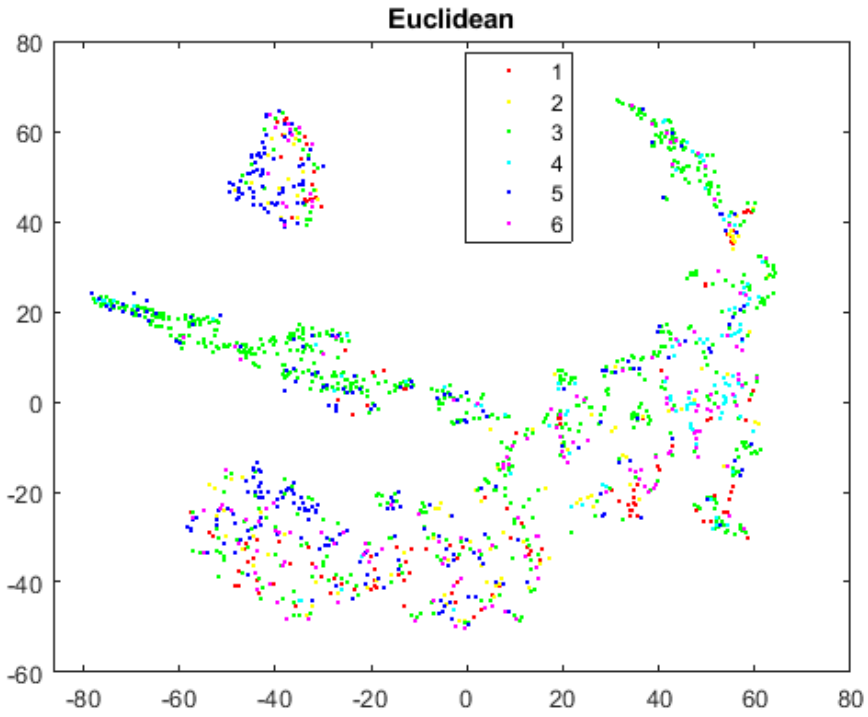


Figure 5.31 t-SNE of 6 class STF features

5.2.2 t-SNE Outputs of 5 Class-Distribution

1 represents “Healthy” class (Red Dots), 2 represents “URTI” class (Yellow Dots), 3 represents “COPD” class (Green Dots), 4 represents “Pneumonia” class (Dark Blue Dots) and 5 represents “Bronchiolitis” class (Purple Dots) in Figures 5.32-5.34 t-SNE outputs. These figures were extracted separately for 3 different features (MFCC, PSD, STF).

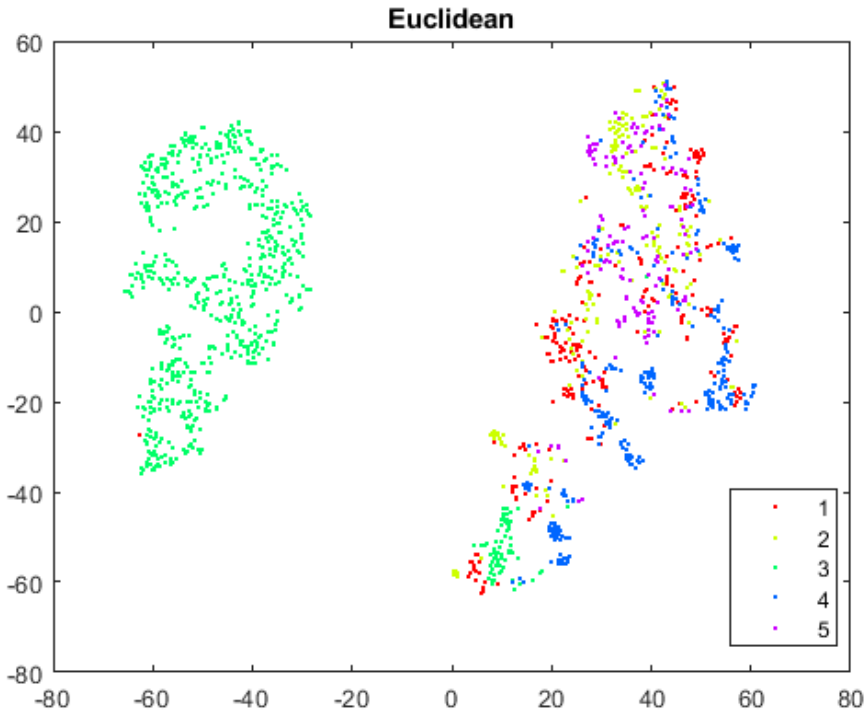


Figure 5.32 t-SNE of 5-class MFCC features

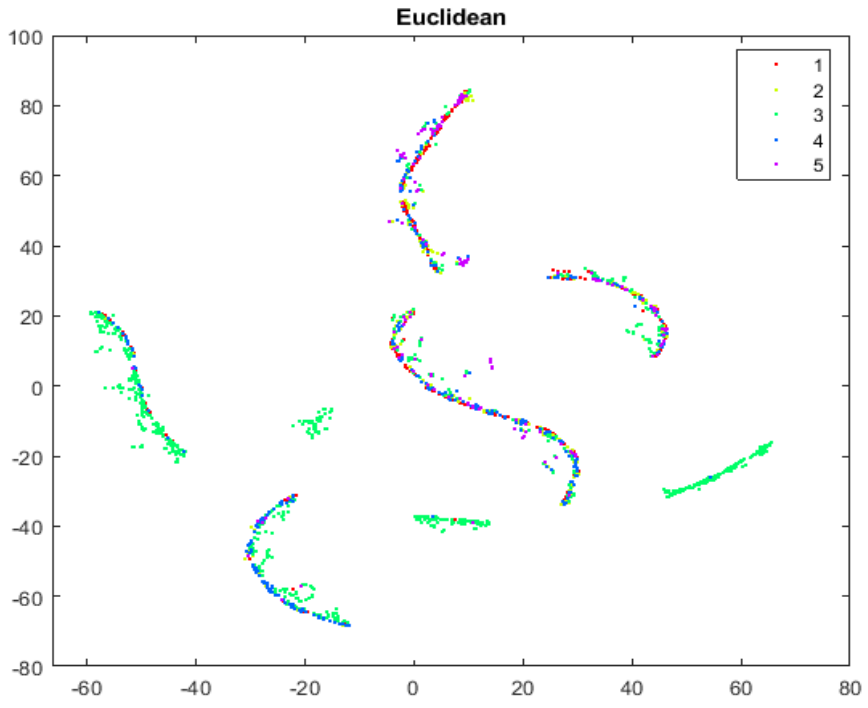


Figure 5.33 t-SNE of 5-Class PSD features

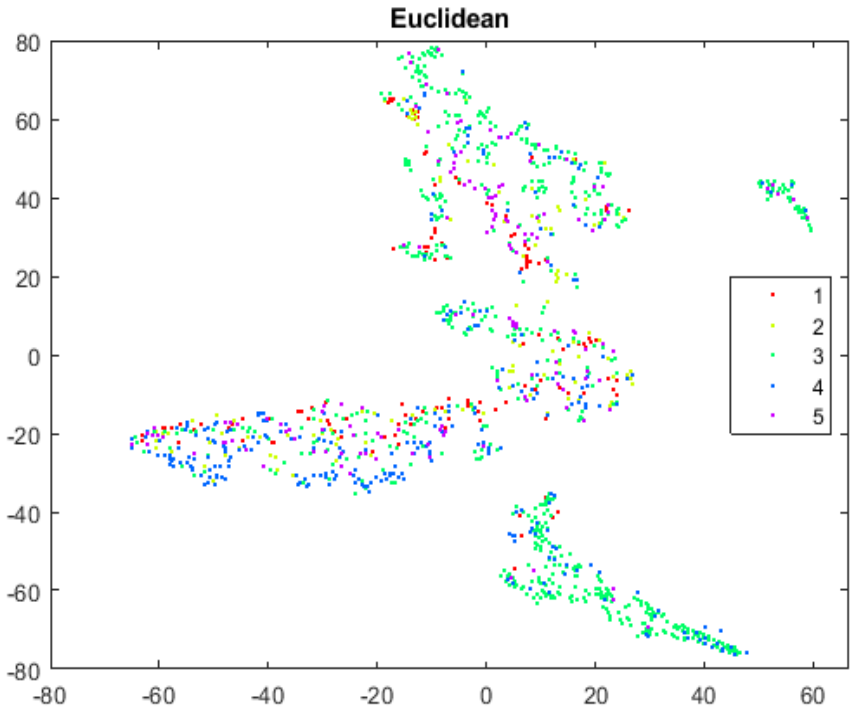


Figure 5.34 t-SNE of 5-class STF features

5.2.3 t-SNE Outputs of 2 Class-Distribution

1 represents “Healthy” class (Red Dots) and 2 represents “Unhealthy” class (Light Blue Dots) in Figures 5.35-5.37 t-SNE outputs. These figures were extracted separately for 3 different features (MFCC, PSD, STF).

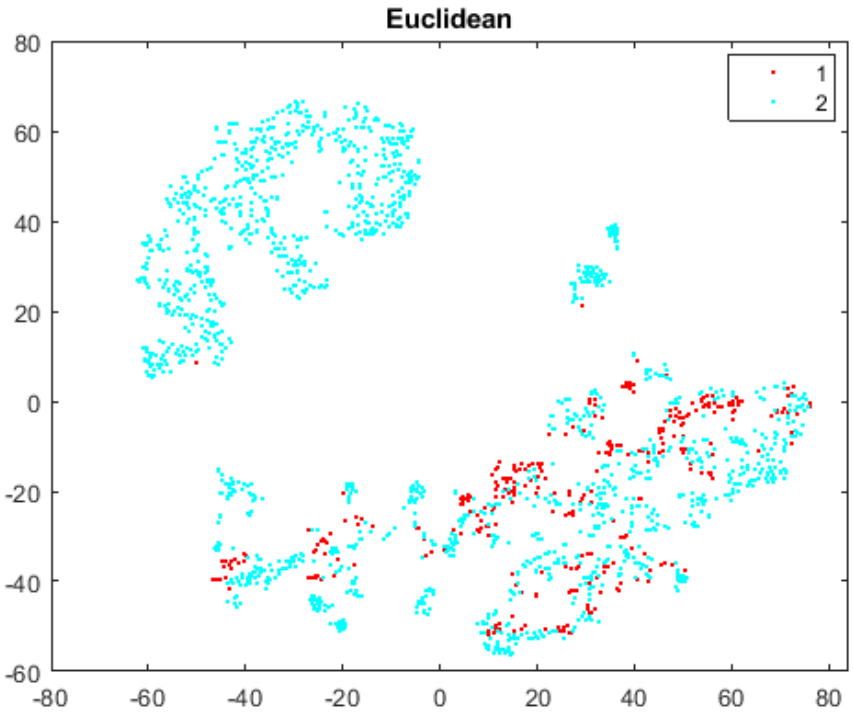


Figure 5.35 t-SNE of 2 Class MFCC features

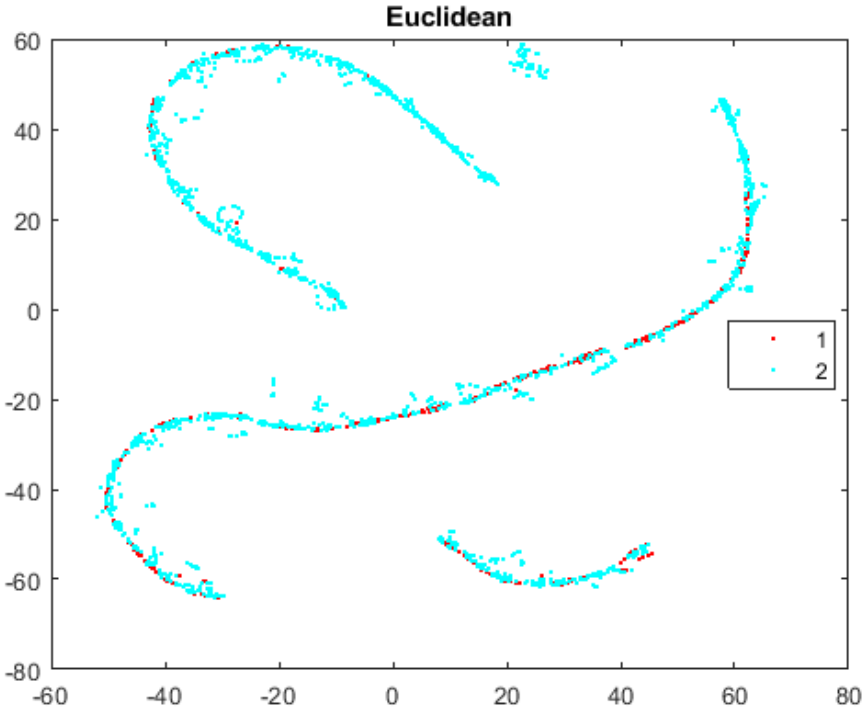


Figure 5.36 t-SNE of 2 Class PSD features

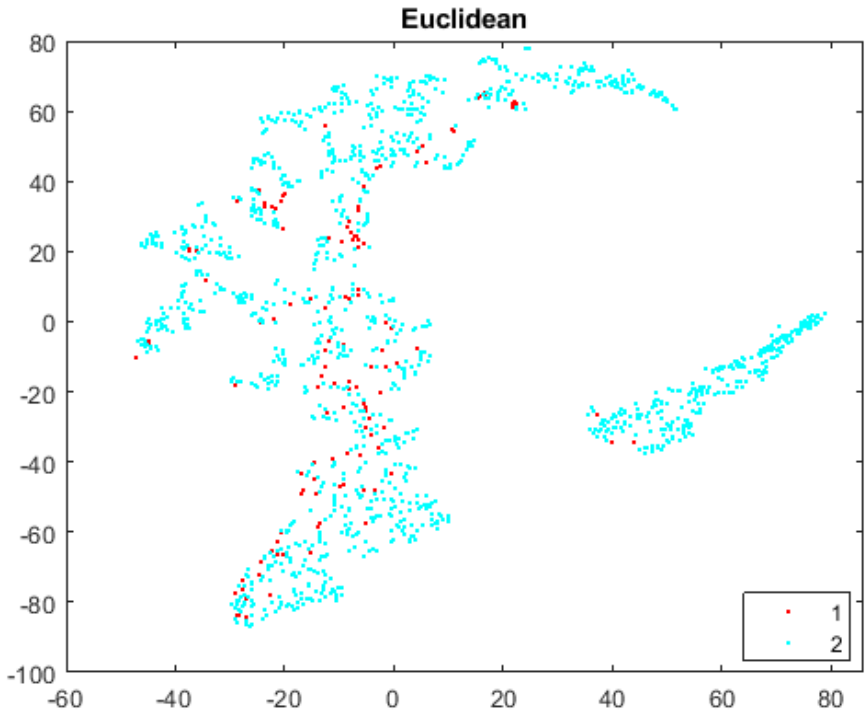


Figure 5.37 t-SNE of 2 Class STF features

5.2.4 Interpretation of t-SNE Outputs

When we look at the t-SNE output of the Two-class (Healthy-Unhealthy) MFCC feature, we see that the Unhealthy (2nd Class) class is separated on the top left side, but on the bottom right, we see that the two classes are mixed with each other. When we look at the t-SNE output of the 6-class MFCC feature, a situation parallel to the interpretation we made in the U-matrix maps emerges. This is the situation where 1st class is mixed with 2nd, 5th and 6th class. Based on t-SNE output of the 5-class MFCC feature, we can say that except for the COPD class (3rd Class:Green), all classes are mixed with each other. In other words, it would be a difficult task to classify most of these classes correctly using MFCC features. Same problems were observed for PSD, and statistical features. When we look at the t-SNE output of the two-class PSD feature, we see that the Unhealthy (2nd Class) class is partially separated at the top, but the two classes are mixed with each other at the bottom. When we look at the t-SNE output of the 6-class PSD feature, we see that some samples of the Bronchectasy

class are partially separated on the top left, but still mixed with the COPD and Healthy classes in the middle side. It is seen that the COPD class distinction is not clear, unlike the MFCC feature, and even if the COPD samples on the right side of the image are separated, the samples on the other side are mixed with all classes. When we look at the t-SNE output of the two-class statistical features, we see that the Unhealthy (2nd Class) class is partially separated at the middle right, lower left and upper middle, but the two classes are mixed with each other on the remaining sides. When we look at the t-SNE output of the 6-class statistical features, we see that all the classes are mixed with each other. After all these t-SNE outputs, we can say that in parallel with the interpretation we made on the U-matrix maps, except for the COPD class, other classes are mixed with each other.

5.3 Density Based Spatial Clustering of Applications with Noise (Dbscan)

Density Based Spatial Clustering of Applications with Noise (Dbscan) algorithm is an algorithm that can visually detect clusters and noise in a dataset (Ester et al., 1996). Eps, one of the two most basic parameters of this algorithm, is a random value equal to or lower than the neighborhood distance between samples according to the Euclidean metric. The other is Minpts, which represents the minimum number of samples with Eps neighborhood to define a cluster. In addition, two important definitions must be known in order to understand this algorithm. The first is density reachability, and the other is density connectivity. If there is a sample chain p1….pn, p1=q, pn=p, we can say that a p point can be density reachable from a q point depending on the Eps and Minpts values (Ester et al., 1996). And a point p is density connected to a point q depending on the Eps and Minpts values if both the p and q points can be density reachable from an o point depending on the Eps and Minpts values (Ester et al., 1996). In addition, according to this algorithm, samples are represented by three different points. If the sample is in any group, it is considered as a core point, if it is on the boundary of any group, it is considered as a border point, and if it is not included in any group, it is considered as a noise (Ester et al., 1996). The algorithm randomly picks a sample p to find a cluster and takes all density reachable points from p according to the Eps and Minpts values (Ester et al., 1996). If

the sample p is a core point, a cluster emerges according to the result (Ester et al., 1996). If this sample is the border point, then there is no density reachable point from p, and the algorithm cannot find a cluster, so it switches to another sample (Ester et al., 1996). If two clusters with different densities are close together and have density-connected common points, they can be combined into a single cluster (Ester et al., 1996). Two sets of points which have the density of the thinnest cluster or denser will be separated from each other if the distance between them is greater than Eps (Ester et al., 1996). Eps and Minpts global values are evaluated according to the thinnest cluster (Ester et al., 1996).

5.3.1 Dbscan Outputs of Healthy Class

Figure 5.38 shows the data distribution of MFCC features of lung audio recordings of healthy people. As seen in the Dbscan output in this figure, the Healthy class is divided into 3 groups. Group -1 (Red Dot) in the figure consists of only one sample. Except for this one sample, the Healthy class consists of two groups, group 1 (Green Dots) and group 2 (Dark Blue Dots). Looking at the number of samples and the distribution of these two groups in Figure 5.38, we can conclude that the distribution of the Healthy class is not homogeneous.

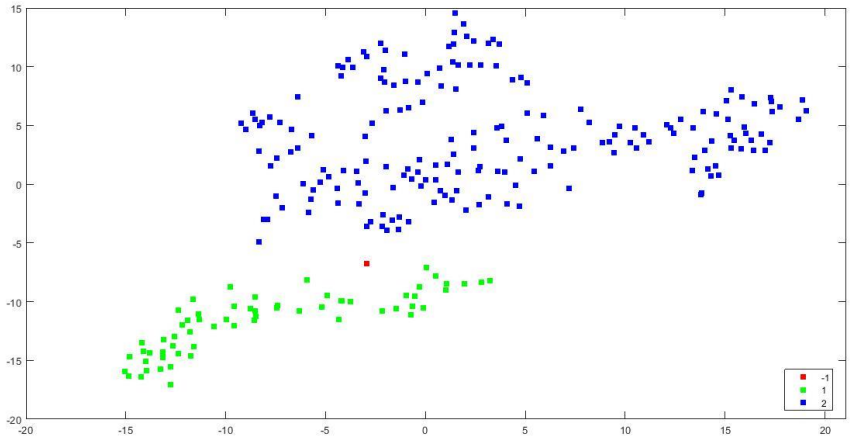


Figure 5.38 Dbscan of healthy class MFCC features

5.3.2 Dbscan Outputs of URTI Class

Figure 5.39 shows the data distribution of the MFCC features of the lung audio recordings of URTI patients. As seen in the Dbscan output in this figure, the URTI class consists of two groups, group 1 (Red Dots) and group 2 (Light Blue Dots). Looking at the number of samples and the distribution of these two groups in Figure 5.39, we can conclude that the distribution of the URTI class is not homogeneous.

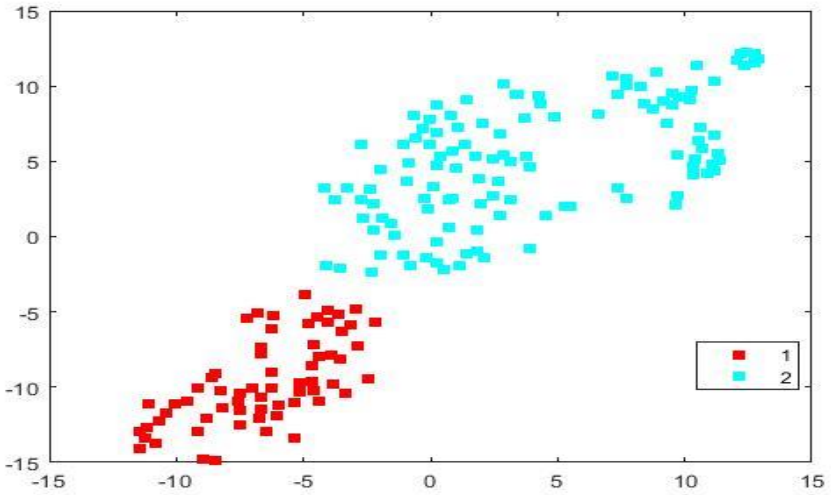


Figure 5.39 Dbscan of URTI class MFCC features

5.3.3 Dbscan Outputs of COPD Class

Figure 5.40 shows the data distribution of the MFCC attributes of the lung audio recordings of COPD patients. As seen in the Dbscan output in this figure, the COPD class consists of two groups, group 1 (Red Dots) and group 2 (Light Blue Dots). From Figure 5.40, we can see that the distribution of the COPD class is close to homogeneous given the small number of samples in group 1 and the distance between the two groups.

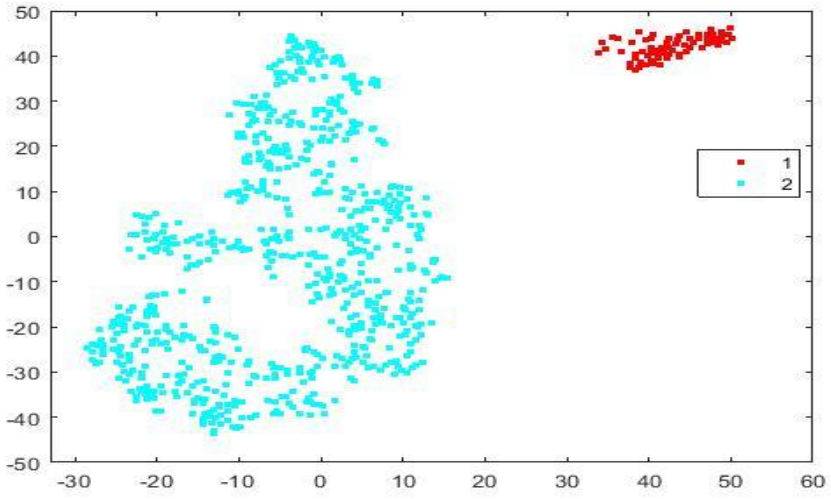


Figure 5.40 Dbscan of COPD class MFCC features

5.3.4 Dbscan Outputs of Pneumonia Class

Figure 5.41 shows the data distribution of the MFCC features of the lung audio recordings of Pneumonia patients. As seen in the Dbscan output in this figure, the Pneumonia class consists of two groups, group 1 (Red Dots) and group 2 (Light Blue Dots). Looking at the number of samples and the distribution of these two groups in Figure 5.41, we can conclude that the distribution of the Pneumonia class is not homogeneous.

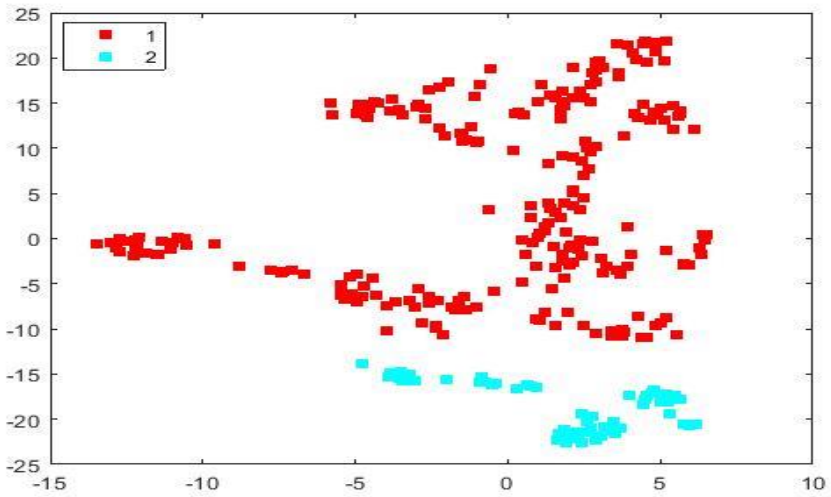


Figure 5.41 Dbscan of Pneumonia class MFCC features

5.3.5 Dbscan Outputs of Bronchiolitis Class

Figure 5.42 shows the data distribution of MFCC features of lung audio recordings of Bronchiolitis patients. As seen in the Dbscan output in this figure, the Bronchiolitis class is divided into 3 groups. Group -1 (Red Dots) in the figure consists of only two samples. Except for these two samples, the Bronchiolitis class consists of two groups, group 1 (Green Dots) and group 2 (Dark Blue Dots). Looking at the number of samples and the distribution of these two groups in Figure 5.42, we can conclude that the distribution of the Bronchiolitis class is not homogeneous.

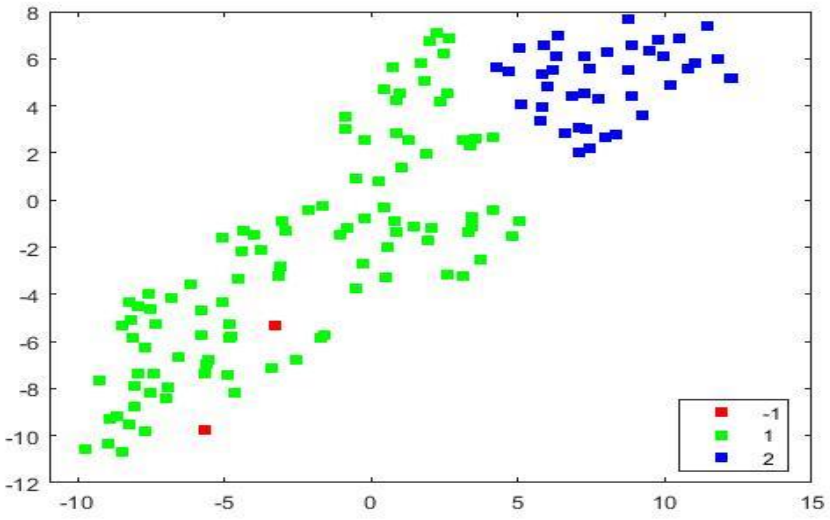


Figure 5.42 Dbscan of Bronchiolitis class MFCC features

5.3.6 Interpretation of Dbscan Outputs

When we evaluated the Dbscan outputs of these 5 classes together with the 5-class t-SNE outputs, we found that some groups of the classes mixed with other classes. When we looked at the LPCC coefficients of some of these mixed samples as mentioned in Section 3, we observed that their values were 0. These samples were deleted from the database both because of their low SNR values (low sound quality).

CHAPTER SIX
METHODS FOR IMBALANCED DATASETS

6.1 Under Sampling Methods

One of the methods used to eliminate the imbalance in the datasets and create a homogeneous dataset is Under Sampling methods. The main purpose of these methods is to reduce the number of samples of the majority class. We used 3 different Under Sampling methods to reduce these numbers.

6.1.1 Random Under Sampling (RUS) Method

The first of the under sampling methods is the Random Under Sampling (RUS) method. In this method, the main purpose is to reduce the number of majority class samples to the desired number. This number is usually equal to the number of minority classes, if there is more than one class, it can be an average value. Reducing the number of samples is made randomly from the majority class (Elhassan et al., 2017). Unfortunately, this may also cause a decrease in the number of samples containing important information from the majority class.

6.1.2 Nearmiss Method

The Nearmiss method is an under sampling method used in imbalance dataset problems. It has three different versions. In the first of these, majority class samples with the lowest average distance from the k nearest neighbors in the minority class are discarded (Mqadi et al., 2021). In the second, on the contrary, majority class samples with the lowest average distance from the k farthest neighbors in the minority class are discarded (Mqadi et al., 2021). In the third, first the k nearest neighbors from the minority class are selected. Afterwards, majority class samples with the highest average distance to these k neighbors are discarded (Mqadi et al., 2021).

6.1.3 Tomek Method

The Tomek method is another under sampling method used in imbalance dataset problems. According to this method, if a sample x in the majority class has a nearest neighbor y in the minority class and the nearest neighbor of y in the minority class is the sample x in the majority class, this majority class sample x is discarded (Elhassan et al., 2017). With this method, the closest examples from the majority class that are likely to be confused with the minority class are discarded. Thus, the number of examples to be misclassified is reduced and the success of the classifier increases.

6.2 Over Sampling Methods

Another method for balancing the distribution of datasets is Over Sampling. The main purpose of these methods is to increase the number of samples of a minority class by adding synthetic data. We used 2 different Over Sampling methods for this process.

6.2.1 Random Over Sampling (ROS) Method

The purpose of Random Over Sampling (ROS) method is to increase the number of minority class samples to the desired number. This number is usually equal to the majority class sample number, it can also be an average value if there is more than one class. Increasing the number of samples is made by multiplying random samples in the minority class (Elhassan et al., 2017). Unfortunately, this may cause the classifier to become over-fitting, reduce its success or increase the computational cost.

6.2.2 Smote Method

The Smote method is another over-sampling method used in imbalance dataset problems. With this method, k nearest neighbors of a random sample in the minority class are found (Elhassan et al., 2017). The distances to these nearest neighbors are multiplied by a random number between 0 and 1, and synthetic data are produced

according to the results obtained. In fact, in this way, the boundaries of the minority class are increased with synthetic data (Elhassan et al., 2017).

6.3 Classification Results of Sampling Method

We ran Decision Tree, Random Forest and XGBoost algorithms by applying 5 different sampling methods to the PSD feature and the PSDnorm feature created by the min-max normalization of this feature. The results of imbalanced methods and machine learning algorithms using Dataset 1 are given in Table 6.1 and Table 6.2.

Table 6.1 Performance values of imbalanced methods

Alg./Feat.	**İmb. Meth.**	**F1**	**Recall**	**MCC**	**Bal. Acc.**	**Tot. Acc.**	**TP**	**TN**	**FN**	**FP**
Decision Tree/PSD	**RUS**	0.7283	0.5833	**0.6157**	**0.7824**	0.7824	63	106	45	2
Random Forest/ PSD	RUS	0.7396	0.6574	0.5508	0.7685	0.7685	71	95	37	13
XGBoost/PSD	RUS	0.6926	0.7407	0.3459	0.6713	0.6713	80	65	28	43
Decision Tree/PSD	ROS	0.6934	0.6571	0.4212	0.7094	0.7094	320	371	167	116
Random Forest/ PSD	ROS	0.7288	0.6345	0.5463	0.7639	0.7639	309	435	178	52
XGBoost/PSD	ROS	0.7042	0.8994	0.2936	0.6222	0.6222	438	168	49	319
Decision Tree/PSD	Tomek	0.7634	0.6404	0.3818	0.7415	0.6791	292	91	164	17
Random Forest/ PSD	Tomek	0.7912	0.6732	0.4391	0.7764	0.7128	307	95	149	13
XGBoost/PSD	**Tomek**	**0.8901**	0.9145	0.3575	0.6609	**0.8174**	417	44	39	64
Decision Tree/PSD	NM	0.2466	0.1667	-0.0243	0.4907	0.4907	18	88	90	20

Table 6.1 Performance values of imbalanced methods (continue)

Random Forest/ PSD	NM	0.6863	0.8611	0.2475	0.6065	0.6065	93	38	15	70
XGBoost/PSD	NM	0.6496	0.7037	0.2442	0.6204	0.6204	76	58	32	50
Decision Tree/PSD	SMO	0.6115	0.4928	0.4055	0.6869	0.6869	240	429	247	58
Random Forest/ PSD	SMO	0.7253	0.6263	0.5464	0.7628	0.7628	305	438	182	49
XGBoost/PSD	SMO	0.7299	0.8768	0.3836	0.6756	0.6756	427	231	60	256
Decision Tree/PSDnorm	RUS	0.5854	0.5556	0.2141	0.6065	0.6065	60	71	48	37
Random Forest/PSDnorm	RUS	0.6728	0.6759	0.3426	0.6713	0.6713	73	72	35	36
XGBoost/PSDnorm	RUS	0.6698	0.6574	0.3521	0.6759	0.6759	71	75	37	33
Decision Tree/PSDnorm	ROS	0.7019	0.8994	0.2856	0.6181	0.6181	438	164	49	323
Random Forest/ PSDnorm	ROS	0.6820	0.8830	0.2186	0.5883	0.5883	430	143	57	344
XGBoost/PSDnorm	ROS	0.7211	0.8789	0.3562	0.6602	0.6602	428	215	59	272
Decision Tree/PSDnorm	Tomek	0.8797	0.8863	0.3394	0.6654	0.8031	413	48	53	60
Random Forest/ PSDnorm	**Tomek**	0.8714	**0.9163**	0.1415	0.5554	0.7805	427	21	39	87
XGBoost/PSDnorm	Tomek	0.8765	0.8906	0.2964	0.6397	0.7962	415	42	51	66
Decision Tree/PSDnorm	NM	0.6207	0.6667	0.1873	0.5926	0.5926	72	56	36	52
Random Forest/ PSDnorm	NM	0.6281	0.7037	0.1717	0.5833	0.5833	76	50	32	58
XGBoost/PSDnorm	NM	0.6260	0.7130	0.1542	0.5741	0.5741	77	47	31	61

Table 6.1 Performance values of imbalanced methods (continue)

Decision Tree/PSDnorm	SMO	0.6751	0.7680	0.2713	0.6304	0.6304	374	240	113	247
Random Forest/ PSDnorm	SMO	0.7556	0.8378	0.4692	0.7290	0.7290	408	302	79	185
XGBoost/PSDnorm	SMO	0.7802	0.8563	0.5276	0.7587	0.7587	417	322	70	165

Table 6.2 Performance values of algorithms using dataset 1

Alg.	**Feature**	**F1**	**Recall**	**MCC**	**Bal. Acc.**	**Tot. Acc.**	**TP**	**TN**	**FN**	**FP**
Decision Tree	PSD (595x43)	0.7713	0.6612	0.3289	0.7102	0.6790	322	82	165	26
Random Forest	PSD (595x43)	0.7654	0.6366	**0.3994**	**0.7581**	0.6807	310	95	177	13
XGBoost	PSD (595x43)	**0.8840**	**0.9076**	0.2798	0.6251	**0.8050**	442	37	45	71

When we compare the results of Dataset 1 and the sampling methods, we see that a significant improvement is achieved with the sampling methods in all performance criteria. If we look at the results obtained by the sampling methods, the Decision Tree algorithm using the PSD feature was the most successful algorithm with the RUS method in MCC and Balanced Accuracy criteria. The Random Forest algorithm using the PSDnorm feature was the most successful algorithm in the Recall criterion with the Tomek method. The XGBoost algorithm using the PSD feature was the most successful algorithm in the F1 and Total Accuracy criteria with the Tomek method.

6.4 Adaptive Synthetic Sampling Approach (ADASYN) Method

Adaptive Synthetic Sampling Approach (ADASYN) algorithm first calculates *d*, the degree of imbalance between n_{min}, the number of the minority class with the low sample number and n_{maj}, the number of the majority class with the high sample number in two-class datasets (He et al., 2008), as seen below.

$$d = n_{\min} / n_{maj} \tag{6.1}$$

If this value is lower than the previously determined class imbalance rate (d_{th}=0.75), the amount of synthetic data to be added to the minority class is calculated as in Formula 6.2 (He et al., 2008).

$$N = \left(n_{maj} - n_{\min}\right) \times \beta \tag{6.2}$$

β value here indicates the desired balance level, so when β=1 we have a fully balanced data set. *K* nearest neighbors are calculated for each minority class sample and the r_j ratio is calculated as in Formula 6.3 (He et al., 2008).

$$r_j = K / \Delta_j \qquad j=1,\ldots n_{min} \tag{6.3}$$

Here, Δ_j value represents the number of nearest neighbors of the minority class sample belonging to the majority class. After that, the r_j's are normalized according to the following formula so that the sum of the $\overline{r_j}$'s equals 1 (He et al., 2008).

$$\overline{r_j} = r_j \Big/ \sum_{j=1}^{n_j} r_j \tag{6.4}$$

The number of synthetic data to be added for each minority sample is determined according to Formula 6.5 (He et al., 2008).

$$n_j = N \times \overline{r_j} \tag{6.5}$$

N value represents the total number of data to be added to the minority class as calculated in Formula 6.2. Finally, n_j synthetic data for each minority class sample x_j are added according to the following procedure: First, a loop is created from 1 to n_j. A random x_{kj} minority data sample is selected among the *K* closest neighbors of the x_j

minority class sample and new synthetic data are created according to Formula 6.6 (He et al., 2008).

$$s_j = x_j + \left(x_{kj} - x_j \right) \times \lambda \tag{6.6}$$

The value of λ is a random coefficient between 0 and 1 (He et al., 2008). When the started loop is finished, n_j synthetic data are created for that minority class sample. The results of the algorithms using the balanced Dataset 2 created with the ADASYN algorithm are seen in Table 6.3.

Table 6.3 Performance values of algorithms using dataset 2

Alg.	**Feature**	**F1**	**Recall**	**MCC**	**Bal. Acc.**	**Tot. Acc.**	**TP**	**TN**	**FN**	**FP**
Decision Tree	PSD (595x43)	0.7885	**0.9648**	0.5420	0.7432	0.7423	466	254	17	233
Random Forest	PSD (595x43)	**0.8064**	0.9441	**0.5838**	**0.7749**	**0.7742**	456	295	27	192
XGBoost	PSD (595x43)	0.5725	0.4534	0.3568	0.6620	0.6629	219	424	264	63

When we compared the results in Dataset 1 in Table 6.2 with the results of our ADASYN algorithm in Table 6.3, we can see that the ADASYN algorithm provides a significant performance increase in 3 other criteria except F1 and Total Accuracy criteria.

6.5 Synthetic Data Generation by Adding Noise

We thougt of balancing the train dataset by adding noise at different SNR values to the healthy train data. When we evaluated the performances of the algorithms by taking into account the different SNR values, we determined that when the noise with an SNR of 38 dB is added to the healthy data, the results are the best. The results of the algorithms using the new dataset (Dataset 3) created by adding 38 dB noise are given in Table 6.4.

Table 6.4 Performance values of algorithms using dataset 3

Alg.	**Feature**	**F1**	**Recall**	**MCC**	**Bal. Acc.**	**Tot. Acc.**	**TP**	**TN**	**FN**	**FP**
Decision Tree	PSD (454x43)	0.4311	0.2803	0.2184	0.6078	0.4361	97	**101**	249	**7**
Decision Tree	MFCC (454x14)	0.7672	0.7572	0.0611	0.5314	0.6498	262	33	84	75
KNN	PSD (454x43)	0.7044	0.6474	0.0322	0.5181	0.5859	224	42	122	66
KNN	MFCC (454x14)	0.7898	0.8092	0.0433	0.5204	0.6718	280	25	66	83
SVM	PSD (454x43)	0.6308	0.4913	0.2398	0.6392	0.5617	170	85	176	23
SVM	MFCC (454x14)	**0.8593**	**0.9884**	-0.0527	0.4942	**0.7533**	342	0	4	108
Random Forest	PSD (454x43)	0.6871	0.5520	**0.3212**	**0.6880**	0.6167	191	89	155	19
Random Forest	MFCC (454x14)	0.8309	0.9017	0.0557	0.5203	0.7203	312	15	34	93
XGBoost	PSD (454x43)	0.8138	0.8526	0.0858	0.5374	0.7026	295	24	51	84
XGBoost	MFCC (454x14)	0.8050	0.8295	0.0960	0.5444	0.6938	287	28	59	80

When we compared the results of Dataset 1 and this method, it was determined that no improvement was achieved in any performance criteria.

6.6 Generative Adversarial Network (GAN)

The Generative Adversarial Network (GAN) algorithm is another synthetic data generation method. The GAN method consists of two artificial neural networks, generative and discriminative. While the generative network tries to reproduce similar ones from the real data given, the discriminative network tries to distinguish the synthetic ones produced with the real data. In the progressive epochs, the generative network begins to produce better synthetic data, while the discriminative network

begins to better distinguish between real and synthetic data. This progress is achieved by using the min max function below (Creswell et al., 2017).

$$\max \min V(G,D) = E_{p_{data}(x)} \log D(x) + E_{p_g(x)} \log(1 - D(G(z))) \tag{6.7}$$

The discriminative network calculates the probability that the data is real with the value *D(x)* (Creswell et al., 2017). The discriminative network tries to make this value a maximum of 1 (Creswell et al., 2017). The generative network, on the other hand, calculates its error in synthetic data with 1-*D(G(z))* value (Creswell et al., 2017). The generative network tries to make this value a minimum of 0 (Creswell et al., 2017). When the optimal value of this function is achieved, the GAN produces synthetic data as close as possible to the real data.

Given that our features were tabular data, we had to utilize Tabular GAN architectures. Numerous Tabular GAN architectures, such as CTGAN (Xu et al., 2019), CWGAN (Engelmann & Lessmann, 2021), CTABGAN (Zhao et al., 2021), and TabFairGAN (Rajabi & Garibay, 2022), have been employed in the literature. We opted for the Conditional Tabular GAN (CTGAN) architecture, particularly suitable for imbalanced datasets (Xu et al., 2019). There is a conditional generator in this GAN architecture (Xu et al., 2019). The conditional generator derives its name from its ability to generate synthetic data based on the conditional distribution of rows in a specific column (Xu et al., 2019). The results of the Random Forest algorithm using the datasets created with the GAN algorithm are shown in Table 6.5.

Table 6.5 Performance values of Random Forest algorithm using GAN datasets

Dataset	**F1**	**Recall**	**MCC**	**Bal. Acc.**	**Tot. Acc.**	**TP**	**TN**	**FN**	**FP**
Dataset 5	0.8604	0.8480	0.2792	0.6462	0.7748	413	48	74	60

Table 6.5 Performance values of Random Forest algorithm using GAN datasets (continue)

Dataset 6	0.7222	0.5791	0.3608	0.7339	0.6353	282	96	205	12
Dataset 7	0.7300	0.5914	0.3560	0.7309	0.6420	288	94	199	14
Dataset 8	0.6890	0.5708	0.1407	0.5910	0.5782	278	66	209	42
Dataset 9	0.8635	0.8706	0.2198	0.6066	0.7748	424	37	63	71
Dataset 10	0.8163	0.7803	0.1662	0.5938	0.7126	380	44	107	64
Dataset 12	0.7946	0.8100	0.7566	0.8100	0.9294	235	318	3	39
Dataset 14	0.7609	0.6468	0.3164	0.7030	0.6672	315	82	172	26
Dataset 15	0.7406	0.7510	0.6847	0.7510	0.9008	228	308	10	49
Dataset 22	0.3361	0.3343	-0.0262	0.3343	0.5042	73	227	165	130
Dataset 23	0.4344	0.4306	0.3352	0.4190	0.5913	31	280	41	174

It would be more accurate to compare the results of the GAN algorithm with the datasets before producing the synthetic data, not among themselves. In this way, we can better observe whether generating synthetic data with GAN improve performance. When evaluated in this way, we can see from Table 6.6 that the results of Dataset 12 are much better in 4 of the 5 criteria compared to the results in the authentic dataset (Dataset 11).

Table 6.6 Performance values of Random Forest algorithm using dataset 12 and authentic dataset

Dataset	**F1**	**Recall**	**MCC**	**Bal. Acc.**	**Tot. Acc.**
Dataset 12 (Synthetic)	**0.7946**	**0.8100**	**0.7566**	**0.8100**	0.9294

Table 6.6: Performance values of Random Forest algorithm using dataset 12 and authentic dataset (continue)

Dataset 11 (Authentic)	0.7296	0.7097	0.7541	0.7097	**0.9546**

We could not obtain the performance increase, achieved in Dataset 12, in other datasets. Although an increase was observed in one or two criteria, the improvement was not achieved in most of the criteria as above.

CHAPTER SEVEN
FEATURE SETS CONSTRUCTED USING SHAPLEY VALUES

7.1 Shapley Value

Shapley value is a method originated from game theory that examines the average contribution of players to a game and is used in a wide range of areas from economics to signal processing (Cavallaro et al., 2021b; Sim et al., 2022; K. Zhang et al., 2022). In particular, we used Shapley values in this study to measure how much the features contribute to the classifier performance (Kababulut et al., 2023). We also benefited the advantages of these values in feature selection and dimension reduction (Kababulut et al., 2023). We used Python's Shapley library in our algorithms to calculate Shapley values as in Equation 7.1 (Shapley, 1951).

$$S_i = \frac{1}{NP} \sum_{Coalitions \not\ni i} \frac{MC_i}{NC} \tag{7.1}$$

where NP represents the number of players, MC_i represents the additional value that a player brings to a coalition when joining, NC represents the number of coalitions excluding the player (Kababulut et al., 2023). In our study, NP corresponds to the number of features, MC_i corresponds to the contributions of the features to the algorithm's performance, and NC corresponds to the number of features except the feature included in the coalitions.

7.2 Feature Sets

We created five datasets with 2-class distribution (Dataset 31-COPD and the others, Dataset 40-Bronchiolitis and the others, Dataset 41-Pneumonia and the others, Dataset 42-URTI and the others, Dataset 43-Healthy and the others). We ran our algorithms in the Python environment, using all features (MFCC+PSD+STF) (Kababulut et al., 2023).

We calculated the Shapley values of the features for the algorithm that we modeled (Kababulut et al., 2023). The order of importance of features according to the Shapley values are shown in Figure 7.1 (Kababulut et al., 2023). Our aim is to increase the performance of our algorithm and reduce the processing load and running time by making both feature selection and dimension reduction using Shapley values (Kababulut et al., 2023). In Figure 7.1, we see the class contribution of the features to the performance according to the Shapley values in the 5-class decision tree algorithm using all (MFCC+PSD+Statistical) features (Kababulut et al., 2023). In this figure, 1 represents Healthy (Magenta), 2 represents URTI (Brown), 3 represents COPD (Purple), 4 represents Pneumonia (Blue) and 5 represents Bronchiolitis (Green) (Kababulut et al., 2023).

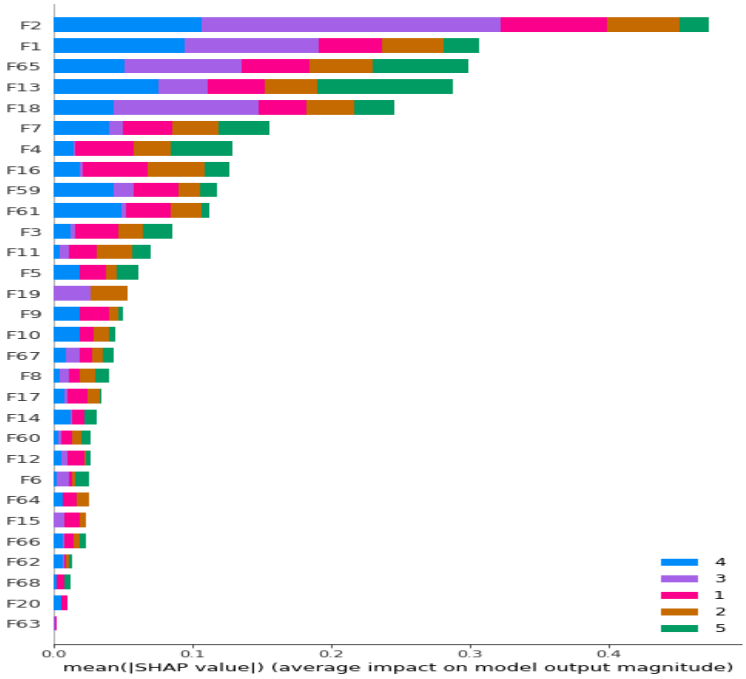


Figure 7.1 Ranking of all 68 features according to their Shapley values for all classes (Kababulut et al., 2023)

In Figure 7.2-7.6, we see the contribution of the features to the classes in the 2-class decision tree algorithm using all features. In Figure 7.2, 1 represents Healthy (Blue) and 2 represents Others (Red), In Figure 7.3, 1 represents URTI (Blue) and 2 represents Others (Red), In Figure 7.4, 1 represents COPD (Red) and 2 represents Others (Blue),

In Figure 7.5, 1 represents Pneumonia (Blue) and 2 represents Others (Red), In Figure 7.6, 1 represents Bronchiolitis (Red) and 2 represents Others (Blue).

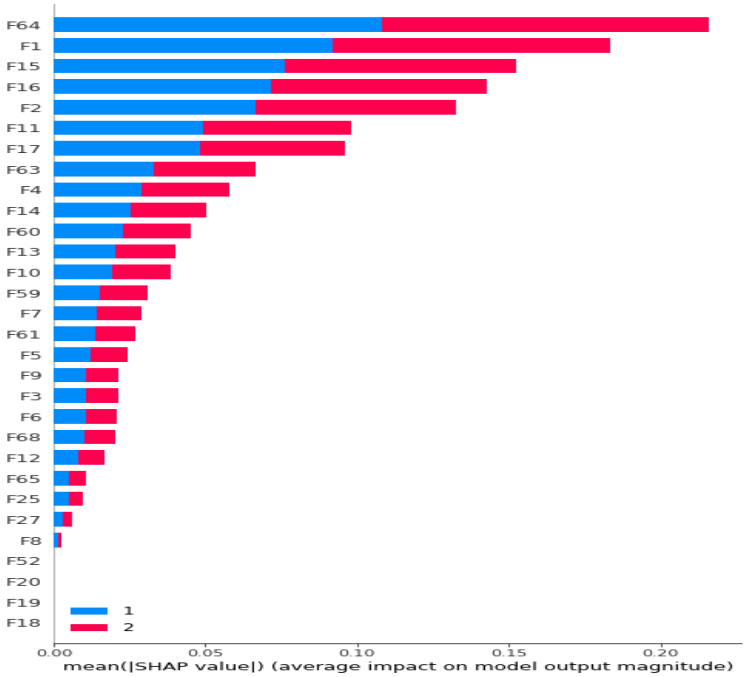


Figure 7.2 Ranking of most contributing 30 features according to their Shapley values for 2-class (Healthy and the Others) classification problem

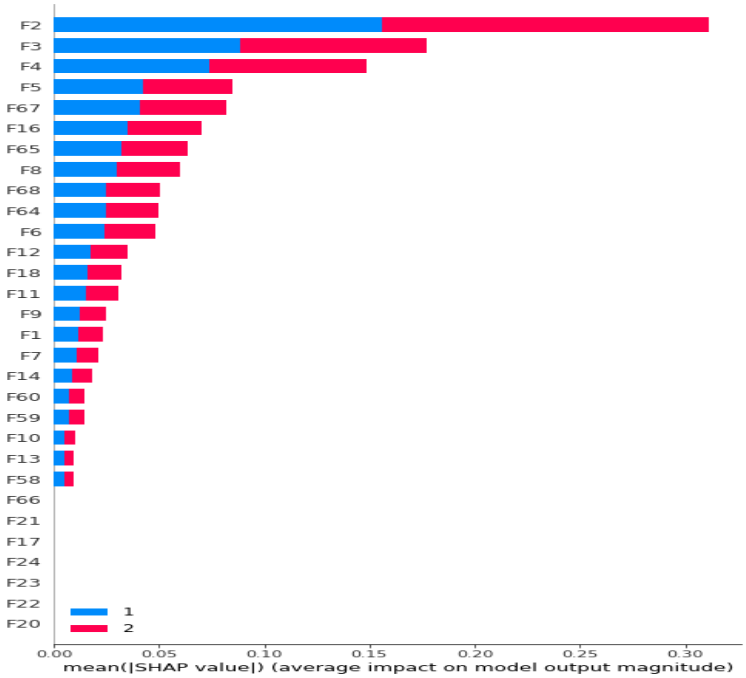


Figure 7.3 Ranking of most contributing 30 features according to their Shapley values for 2-class (URTI and the Others) classification problem (Kababulut et al., 2023)

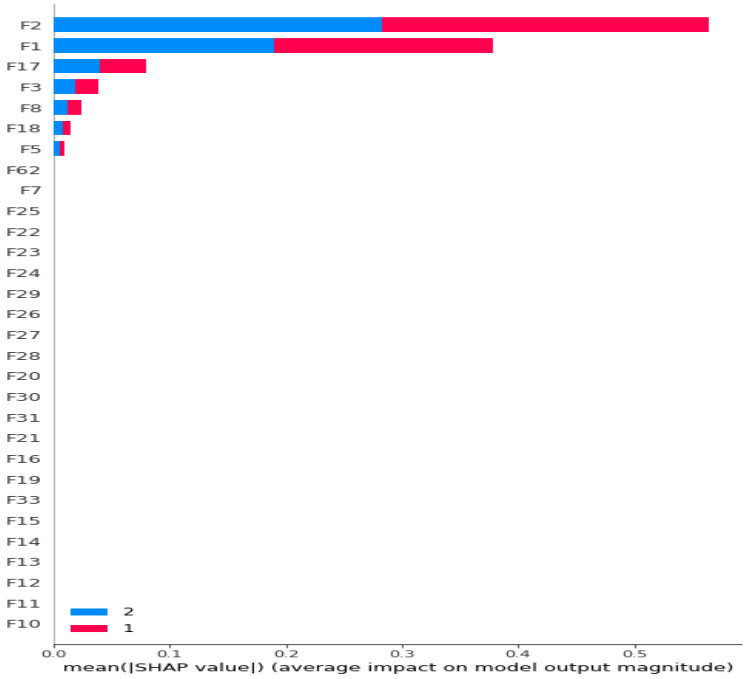


Figure 7.4 Ranking of most contributing 30 features according to their Shapley values for 2-class (COPD and the Others) classification problem

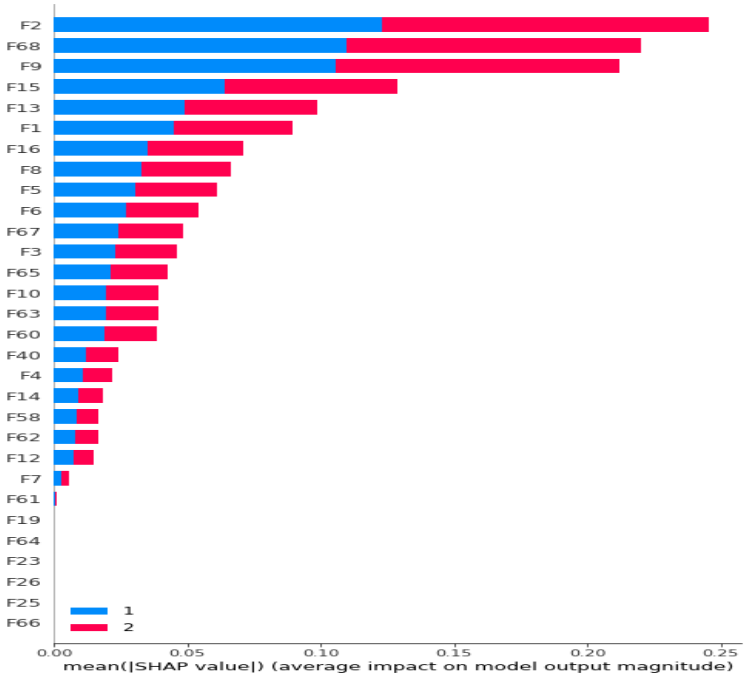


Figure 7.5 Ranking of most contributing 30 features according to their Shapley values for 2-class (Pneumonia and the Others) classification problem

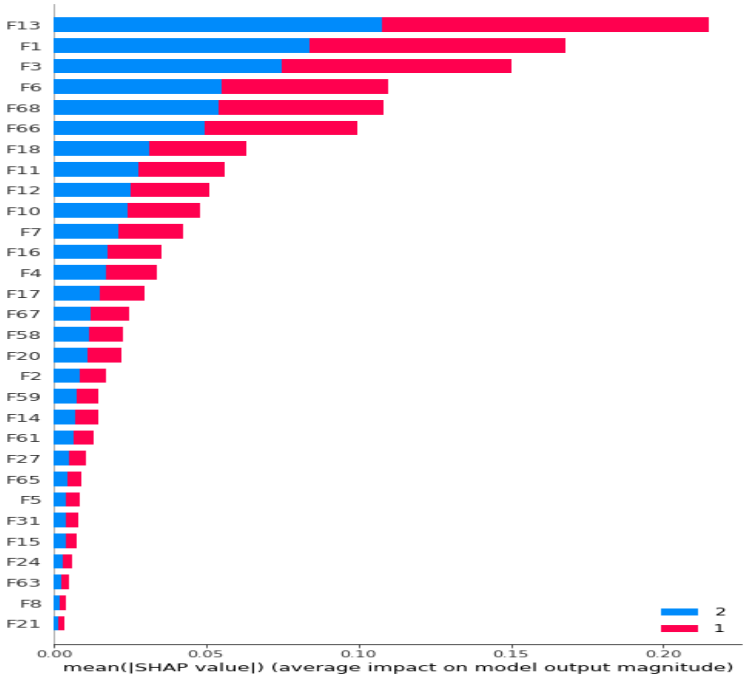


Figure 7.6 Ranking of most contributing 30 features according to their Shapley values for 2-class (Bronchiolitis and the Others) classification problem

7.2.1 Feature Set 1

In this study, 5-class lung disease classification is made with the Decision Tree Algorithm (DTA) as the classifier using different feature sets (Kababulut et al., 2023). For this purpose, binary and 5-class disease classification was made using all the extracted features (i.e., MFCC, PSD, and Statistical features) (Kababulut et al., 2023). 14 MFCC, 43 PSD, and 11 STF features, in total 68 features are used (Kababulut et al., 2023). This feature set is called as Feature Set 1. Using this feature set in classification is named as the first classification method (Method 1) (Kababulut et al., 2023).

7.2.2 Feature Set 2

Based on Figure 7.1, in this feature set only the order of importance of the features (contribution to performance) was considered (Kababulut et al., 2023). 30 (2,1,65,13,18,… and 63) features were ranked according to their contributions. This way, we did not try all possible choices. The performance of the DTA (binary and 5-

class disease classification) was examined by using these features individually. This feature set is called as Feature Set 2. Feature Set 2 was used in the second classification method (Method 2). The absolute Shapley values of the 30 features shown in Figure 7.1 are in Table 7.1. This table shows the Shapley values of the features on class basis as well as the total absolute Shapley values in the last column. In fact, the ranking in Figure 7.1 is based on these total values.

Table 7.1 Absolute Shapley values of feature set 2

Feature Number	**Class 1**	**Class 2**	**Class 3**	**Class 4**	**Class 5**	**Total**
F2	0.0772	0.0514	0.2154	0.1069	0.0218	**0.4727**
F1	0.0453	0.0447	0.0969	0.0940	0.0263	**0.3072**
F65	0.0490	0.0454	0.0841	0.0511	0.0689	**0.2985**
F13	0.0412	0.0379	0.0362	0.0751	0.0976	**0.2880**
F18	0.0345	0.0350	0.1047	0.0430	0.0282	**0.2454**
F7	0.0355	0.0333	0.0095	0.0401	0.0373	**0.1557**
F4	0.0418	0.0273	0.0009	0.0144	0.0441	**0.1286**
F16	0.0462	0.0417	0.0030	0.0183	0.0171	**0.1262**
F59	0.0319	0.0164	0.0148	0.0429	0.0118	**0.1179**
F61	0.0327	0.0221	0.0025	0.0492	0.0062	**0.1127**
F3	0.0314	0.0181	0.0034	0.0120	0.0209	**0.0859**
F11	0.0198	0.0257	0.0071	0.0042	0.0132	**0.0700**
F5	0.0190	0.0078	0.0004	0.0184	0.0150	**0.0605**

Table 7.1 Absolute Shapley values of feature set 2 (continue)

F19	0.0007	0.0268	0.0261	0.0000	0.0000	**0.0536**
F9	0.0205	0.0072	0.0006	0.0184	0.0037	**0.0504**
F10	0.0101	0.0108	0.0000	0.0192	0.0042	**0.0442**
F67	0.0094	0.0070	0.0103	0.0083	0.0078	**0.0429**
F8	0.0077	0.0119	0.0058	0.0049	0.0099	**0.0401**
F17	0.0149	0.0088	0.0019	0.0081	0.0012	**0.0348**
F14	0.0092	0.0000	0.0015	0.0116	0.0085	**0.0309**
F60	0.0075	0.0063	0.0028	0.0030	0.0069	**0.0265**
F12	0.0123	0.0011	0.0044	0.0053	0.0033	**0.0263**
F6	0.0025	0.0017	0.0093	0.0020	0.0103	**0.0257**
F64	0.0107	0.0080	0.0000	0.0064	0.0000	**0.0251**
F15	0.0117	0.0042	0.0074	0.0000	0.0003	**0.0236**
F66	0.0066	0.0049	0.0012	0.0066	0.0041	**0.0233**
F62	0.0015	0.0024	0.0008	0.0068	0.0021	**0.0136**
F68	0.0058	0.0000	0.0000	0.0022	0.0038	**0.0118**
F20	0.0052	0.0000	0.0000	0.0052	0.0000	**0.0105**
F63	0.0010	0.0000	0.0010	0.0000	0.0000	**0.0020**

7.2.3 Feature Sets 3

Considering Figure 7.1, in these feature sets 30 features that make the highest contribution to separating five classes were selected in descending contribution order (Kababulut et al., 2023). In Method 3, for example, for Dataset 40, 30 features (13,65,4,7,18,… and 63) that contributed the most to the Bronchiolitis class (Class 5 in green in this graph) were created. Thus, individual feature sets are extracted for 5 classes and the performance of the five different DTAs were examined for each separate class. These feature sets are called as Feature Sets 3. These feature sets are the datasets we prepared, taking into account the *class contribution*. Feature Sets 3 were used in the third classification method (Method 3). The ranking of the features with high Shapley values in Figure 7.1 on class basis is shown in Table 7.2. In this table, the features ranked for the classes also become the set of features for the datasets. The Shapley values in this table are calculated based on the contribution of the features to the classification performance of the five-class DTA.

Table 7.2 Shapley values of feature sets 3 on class basis

Dataset 43 (H/O) Class 1		**Dataset 42 (U/O) Class 2**		**Dataset 31 (C/O) Class 3**		**Dataset 41 (P/O) Class 4**		**Dataset 40 (B/O) Class 5**	
Feature	**Shapley Value**	**Feature**	**Shapley Value**	**Feature**	**Shapley Value**	**Feature**	**Shapley Value**	**Feature**	**Shapley Value**
F2	0.0772	**F2**	0.0514	**F2**	0.2154	**F2**	0.1069	**F13**	0.0976
F65	0.0490	**F65**	0.0454	**F18**	0.1047	**F1**	0.0940	**F65**	0.0689
F16	0.0462	**F1**	0.0447	**F1**	0.0969	**F13**	0.0751	**F4**	0.0441
F1	0.0453	**F16**	0.0417	**F65**	0.0841	**F65**	0.0511	**F7**	0.0373
F4	0.0418	**F13**	0.0379	**F13**	0.0362	**F61**	0.0492	**F18**	0.0282
F13	0.0412	**F18**	0.0350	**F19**	0.0261	**F18**	0.0430	**F1**	0.0263

Table 7.2 Shapley values of feature sets 3 on class basis (continue)

F7	0.0355	**F7**	0.0333	**F59**	0.0148	**F59**	0.0429	**F2**	0.0218
F18	0.0345	**F4**	0.0273	**F67**	0.0103	**F7**	0.0401	**F3**	0.0209
F61	0.0327	**F19**	0.0268	**F7**	0.0095	**F10**	0.0192	**F16**	0.0171
F59	0.0319	**F11**	0.0257	**F6**	0.0093	**F9**	0.0184	**F5**	0.0150
F3	0.0314	**F61**	0.0221	**F15**	0.0074	**F5**	0.0184	**F11**	0.0132
F9	0.0205	**F3**	0.0181	**F11**	0.0071	**F16**	0.0183	**F59**	0.0118
F11	0.0198	**F59**	0.0164	**F8**	0.0058	**F4**	0.0144	**F6**	0.0103
F5	0.0190	**F8**	0.0119	**F12**	0.0044	**F3**	0.0120	**F8**	0.0099
F17	0.0149	**F10**	0.0108	**F3**	0.0034	**F14**	0.0116	**F14**	0.0085
F12	0.0123	**F17**	0.0088	**F16**	0.0030	**F67**	0.0083	**F67**	0.0078
F15	0.0117	**F64**	0.0080	**F60**	0.0028	**F17**	0.0081	**F60**	0.0069
F64	0.0107	**F5**	0.0078	**F61**	0.0025	**F62**	0.0068	**F61**	0.0062
F10	0.0101	**F9**	0.0072	**F17**	0.0019	**F66**	0.0066	**F10**	0.0042
F67	0.0094	**F67**	0.0070	**F14**	0.0015	**F64**	0.0064	**F66**	0.0041
F14	0.0092	**F60**	0.0063	**F66**	0.0012	**F12**	0.0053	**F68**	0.0038
F8	0.0077	**F66**	0.0049	**F63**	0.0010	**F20**	0.0052	**F9**	0.0037
F60	0.0075	**F15**	0.0042	**F4**	0.0009	**F8**	0.0049	**F12**	0.0033
F66	0.0066	**F62**	0.0024	**F62**	0.0008	**F11**	0.0042	**F62**	0.0021

Table 7.2 Shapley values of feature sets 3 on class basis (continue)

F68	0.0058	**F6**	0.0017	**F9**	0.0006	**F60**	0.0030	**F17**	0.0012
F20	0.0052	**F12**	0.0011	**F5**	0.0004	**F68**	0.0022	**F15**	0.0003
F6	0.0025	**F14**	0.0000	**F10**	0.0000	**F6**	0.0020	**F19**	0.0000
F62	0.0015	**F68**	0.0000	**F64**	0.0000	**F15**	0.0000	**F64**	0.0000
F63	0.0010	**F20**	0.0000	**F68**	0.0000	**F19**	0.0000	**F20**	0.0000
F19	0.0007	**F63**	0.0000	**F20**	0.0000	**F63**	0.0000	**F63**	0.0000

7.2.4 Feature Sets 4

In these feature sets, the selection was made according to the direct ordering of the features, considering Figures 7.2-7.6 separately (Kababulut et al., 2023). In other words, in binary classifications, considering only the order of importance of the features, 30 (For example, 13,1,3,6,68,… and 21 in the Dataset 40 -Bronchiolitis and the others- in Figure 7.6) features were created. Thus, individual feature sets are extracted for 5 classes and the performance of the 5 different DTAs were examined for each separate class. These feature sets are called as Feature Sets 4. These feature sets are the datasets we prepared, taking into account the *class contribution*. Feature Sets 4 were used in the fourth classification method (Method 4). The absolute Shapley values of the 30 features seen in Figures 7.2-7.6 are in Table 7.3. In this table, the features listed for each classification become a set of features for the datasets. The Shapley values in this table are calculated based on the contribution of the features to the classification performance of the two-class DTAs.

Table 7.3 Absolute Shapley values of feature sets 4 in binary classifications

Dataset 43 (H/O)	**Dataset 42 (U/O)**	**Dataset 31 (C/O)**	**Dataset 41 (P/O)**	**Dataset 40 (B/O)**

Table 7.3 Absolute Shapley values of feature sets 4 in binary classifications (continue)

Feature	Shapley Value	Feature	Shapley Value	Feature	Shapley Value	Feature	Shapley Value	Feature	Shapley Value
F64	0.1081	**F2**	0.1557	**F2**	0.2823	**F2**	0.1228	**F13**	0.1076
F1	0.0918	**F3**	0.0887	**F1**	0.1895	**F68**	0.1099	**F1**	0.0840
F15	0.0761	**F4**	0.0742	**F17**	0.0396	**F9**	0.1059	**F3**	0.0750
F16	0.0715	**F5**	0.0426	**F3**	0.0192	**F15**	0.0644	**F6**	0.0550
F2	0.0664	**F67**	0.0408	**F8**	0.0120	**F13**	0.0493	**F68**	0.0541
F11	0.0491	**F16**	0.0351	**F18**	0.0075	**F1**	0.0447	**F66**	0.0497
F17	0.0480	**F65**	0.0318	**F5**	0.0048	**F16**	0.0354	**F18**	0.0315
F63	0.0332	**F8**	0.0299	**F62**	0.0000	**F8**	0.0331	**F11**	0.0280
F4	0.0290	**F68**	0.0251	**F7**	0.0000	**F5**	0.0305	**F12**	0.0254
F14	0.0252	**F64**	0.0247	**F25**	0.0000	**F6**	0.0270	**F10**	0.0241
F60	0.0227	**F6**	0.0243	**F22**	0.0000	**F67**	0.0243	**F7**	0.0212
F13	0.0201	**F12**	0.0175	**F23**	0.0000	**F3**	0.0230	**F16**	0.0178
F10	0.0192	**F18**	0.0161	**F24**	0.0000	**F65**	0.0213	**F4**	0.0171
F59	0.0155	**F11**	0.0153	**F29**	0.0000	**F10**	0.0196	**F17**	0.0149
F7	0.0144	**F9**	0.0125	**F26**	0.0000	**F63**	0.0195	**F67**	0.0123
F61	0.0135	**F1**	0.0116	**F27**	0.0000	**F60**	0.0192	**F58**	0.0114
F5	0.0123	**F7**	0.0107	**F28**	0.0000	**F40**	0.0121	**F20**	0.0110

Table 7.3 Absolute Shapley values of feature sets 4 in binary classifications (continue)

F9	0.0108	**F14**	0.0090	**F20**	0.0000	**F4**	0.0108	**F2**	0.0085
F3	0.0105	**F60**	0.0075	**F30**	0.0000	**F14**	0.0093	**F59**	0.0073
F6	0.0104	**F59**	0.0074	**F31**	0.0000	**F58**	0.0085	**F14**	0.0072
F68	0.0101	**F10**	0.0051	**F21**	0.0000	**F62**	0.0083	**F61**	0.0066
F12	0.0083	**F13**	0.0047	**F16**	0.0000	**F12**	0.0074	**F27**	0.0052
F65	0.0052	**F58**	0.0047	**F19**	0.0000	**F7**	0.0030	**F65**	0.0044
F25	0.0049	**F66**	0.0000	**F33**	0.0000	**F61**	0.0005	**F5**	0.0042
F27	0.0031	**F21**	0.0000	**F15**	0.0000	**F19**	0.0000	**F31**	0.0039
F8	0.0013	**F17**	0.0000	**F14**	0.0000	**F64**	0.0000	**F15**	0.0039
F52	0.0000	**F24**	0.0000	**F13**	0.0000	**F23**	0.0000	**F24**	0.0031
F20	0.0000	**F23**	0.0000	**F12**	0.0000	**F26**	0.0000	**F63**	0.0024
F19	0.0000	**F22**	0.0000	**F11**	0.0000	**F25**	0.0000	**F8**	0.0020
F18	0.0000	**F20**	0.0000	**F10**	0.0000	**F66**	0.0000	**F21**	0.0017

7.2.5 Feature Sets 5

In these feature sets, the class percentages of each of the features in Figure 7.1 were calculated. Then, for each class, the features with the highest percentages were ranked and feature selection was performed with the relative Shapley values of the features, which has not been seen in any study in the literature before. For example, the 30 highest percentage features for Dataset 40 are ranked as 6, 4, 13, 68, 14, ... and 63. The feature sets extracted in this way for each class are called Feature Sets 5. These feature

sets are the datasets we prepared, taking into account the *class contribution*. Feature Sets 5 were used in the fifth classification method (Method 5). With these feature sets, the performance values of DTA (Method 5), which performs 5 binary classifications, were calculated. The features in Figure 7.1 are ranked as shown according to the high percentage of Shapley values per class as shown in Table 7.4. For instance, because Feature 20 (F20) contributes equally (Shapley values are equal) only to the pneumonia and healthy classes, this feature has a 50 percent contribution for both classes. The features ranked for the classes in this table also become a set of features for the datasets. The Shapley values in this table are calculated based on the contribution of the features to the classification performance of the five-class DTA.

Table 7.4 Percentage of Shapley values of feature sets 5

Dataset 43 (H/O) Class 1		**Dataset 42 (U/O) Class 2**		**Dataset 31 (C/O) Class 3**		**Dataset 41 (P/O) Class 4**		**Dataset 40 (B/O) Class 5**	
Feat.	**Percentage of Shapley Value**	**Feat.**	**Percentage of Shapley Value**	**Feat.**	**Percentage of Shapley Value**	**Feat.**	**Percentage of Shapley Value**	**Feat.**	**Percentage of Shapley Value**
F20	50.00%	**F19**	50.00%	**F63**	50.00%	**F20**	50.00%	**F6**	39.87%
F63	50.00%	**F11**	36.75%	**F19**	48.70%	**F62**	49.82%	**F4**	34.25%
F15	49.46%	**F16**	33.04%	**F2**	45.57%	**F61**	43.64%	**F13**	33.88%
F68	49.23%	**F64**	31.81%	**F18**	42.65%	**F10**	43.46%	**F68**	32.16%
F12	46.60%	**F8**	29.65%	**F6**	36.14%	**F14**	37.64%	**F14**	27.50%
F17	42.84%	**F17**	25.23%	**F15**	31.57%	**F9**	36.53%	**F60**	26.22%
F64	42.76%	**F10**	24.33%	**F1**	31.53%	**F59**	36.38%	**F5**	24.75%
F9	40.67%	**F60**	23.74%	**F65**	28.19%	**F1**	30.60%	**F8**	24.63%
F3	36.62%	**F7**	21.36%	**F67**	24.00%	**F5**	30.32%	**F3**	24.37%

Table 7.4 Percentage of Shapley values of feature sets 5 (continue)

F16	36.58%	**F4**	21.26%	**F12**	16.75%	**F66**	28.22%	**F7**	23.93%
F4	32.54%	**F3**	21.12%	**F8**	14.40%	**F13**	26.09%	**F65**	23.08%
F5	31.45%	**F66**	21.00%	**F13**	12.59%	**F7**	25.79%	**F11**	18.87%
F14	29.92%	**F61**	19.64%	**F59**	12.55%	**F64**	25.44%	**F67**	18.28%
F61	29.02%	**F15**	17.88%	**F60**	10.41%	**F17**	23.23%	**F66**	17.48%
F60	28.32%	**F62**	17.86%	**F11**	10.17%	**F2**	22.62%	**F62**	15.25%
F66	28.32%	**F67**	16.40%	**F62**	6.18%	**F12**	20.04%	**F16**	13.52%
F11	28.25%	**F65**	15.20%	**F7**	6.09%	**F67**	19.40%	**F12**	12.50%
F59	27.09%	**F1**	14.56%	**F17**	5.33%	**F68**	18.61%	**F18**	11.48%
F7	22.82%	**F9**	14.30%	**F66**	4.99%	**F18**	17.53%	**F59**	10.04%
F10	22.75%	**F18**	14.26%	**F14**	4.93%	**F65**	17.11%	**F10**	9.46%
F67	21.92%	**F59**	13.94%	**F3**	3.90%	**F16**	14.51%	**F1**	8.56%
F8	19.11%	**F13**	13.15%	**F16**	2.34%	**F3**	13.99%	**F9**	7.40%
F65	16.42%	**F5**	12.89%	**F61**	2.21%	**F8**	12.21%	**F61**	5.49%
F2	16.32%	**F2**	10.87%	**F9**	1.11%	**F60**	11.31%	**F2**	4.62%
F1	14.75%	**F6**	6.64%	**F4**	0.73%	**F4**	11.23%	**F17**	3.37%
F13	14.29%	**F12**	4.11%	**F5**	0.59%	**F6**	7.71%	**F15**	1.08%
F18	14.07%	**F14**	0.00%	**F10**	0.00%	**F11**	5.96%	**F19**	0.00%

Table 7.4 Percentage of Shapley values of feature sets 5 (continue)

F62	10.89%	**F68**	0.00%	**F64**	0.00%	**F15**	0.00%	**F64**	0.00%
F6	9.65%	**F20**	0.00%	**F68**	0.00%	**F19**	0.00%	**F20**	0.00%
F19	1.30%	**F63**	0.00%	**F20**	0.00%	**F63**	0.00%	**F63**	0.00%

CHAPTER EIGHT
PROPOSED CLASSIFICATION METHODS AND SIMULATIONS RESULTS

8.1 Evaluation Metrics

In this study. we would like to classify whether a person has any lung disease and its type from only lung sounds. We evaluated the successes of the classifiers with five different performance criteria. The first of these is the Total Accuracy (*Tacc*) performance criterion. *Tacc* is calculated as in Equation 8.1 as the ratio of the number of true positive to the total number of samples.

$$Tacc = \frac{TP}{TP + FN + TN + FP} \tag{8.1}$$

TP is correct predictions in the class labeled positive. FN is false predictions in the positive class. TN is correct predictions in the class labeled negative, FP is false predictions in the negative class. Balanced Accuracy (*Bacc*) is one of the most widely used criteria for classifying unbalanced datasets. As can be seen in Formula 8.2, a balanced accuracy value is calculated by giving equal weight to each class.

$$Bacc = \frac{1}{N}\sum_{i=1}^{N}\frac{TP_i}{TP_i + FN_i} \tag{8.2}$$

TP_i is correct predictions in the i*th* class, FN_i is false predictions in the i*th* class and N is the number of classes. Another performance criterion is the F1 score. F1 score, which is the harmonic mean of precision and recall, is a criterion that shows test accuracy and is calculated as follows:

$$F1 = \frac{TP}{(TP + \frac{1}{2}(FP + FN))} \tag{8.3}$$

Recall is a criterion that shows how sensitively a classifier classifies all elements in the class that are characterized as positive. Recall, also called True Positive Rate, is calculated as in Equation 8.4.

$$\mathrm{Re}\,call = \frac{TP}{TP + FN} \tag{8.4}$$

The fifth performance criterion is Matthew Correlation Coefficient (MCC). This criterion is a more reliable criterion than other criteria especially for imbalanced datasets, which can get a high value if a classifier has successful results in true positive and true negative (Chicco et al., 2021). MCC, which can take values between -1 and 1, is calculated as in Equation 8.5.

$$MCC = \frac{TP.TN - FP.FN}{\sqrt{(TP + FP).(TP + FN).(TN + FP).(TN + FN)}} \tag{8.5}$$

8.2 Binary Classifications

In this section, the two classes of datasets detailed in Chapter 3 were classified. These classifications were performed in two different ways, without and with the use of Shapley's methods. In the classifications without Shapley's methods, the performance values of 4 different machine learning algorithms were calculated by separating the features of different patient records as train-test. In the classifications with Shapley methods, only the 5-Fold Cross Validation performance values of the Decision Tree algorithm were calculated.

8.2.1 Binary Classifications without Shapley Methods

In this section, binary classifications were performed for all datasets except Dataset 31, 40, 41, 42 and 43, which are the two-class datasets described in Chapter 3. In these classifications, the performance results of different machine learning algorithms (Random Forest-RF, XGBoost-XG, K Nearest Neigbor-K-NN and Decision Tree-DT)

in 5 different metrics were calculated using three features (MFCC, PSD and STF). These results were then compared with each other to determine the best performing datasets, algorithms and features. Shapley's Methods described in Chapter 7 were not used in these classifications.

8.2.1.1 Healthy Class Binary Classifications

Dataset 4 is a dataset labeled as Healthy vs. Unhealthy (URTI, COPD, Bronchectasy, Pneumonia and Bronchiolitis) in which MFCC features are divided into 500 msec segments. Dataset 16 is a dataset labeled as Healthy vs. Unhealthy (URTI, COPD, Bronchectasy, Pneumonia and Bronchiolitis) where records with LPCC values of zero were deleted. As seen in Table 8.1 and 8.2, the Decision Tree algorithm achieved the most successful results in Dataset 16 with the PSD features in 3 criteria (F1, MCC and Balanced Accuracy). The Random Forest algorithm obtained the most successful results in Recall criterion with PSD features and in Total Accuracy criterion with MFCC features in Dataset 16. Also, as can be seen from Table 8.2, Support Vector Machine (SVM), Neural Network (NN) and Logistic Regression (LR) algorithms failed to classify the healthy class, so we did not use these three algorithms in any of the subsequent classifications.

Table 8.1 Performance values of Random Forest algorithm using binary healthy class datasets

Dataset/Feat.	**F1**	**Recall**	**MCC**	**Bal. Acc.**	**Tot. Acc.**	**TP**	**TN**	**FN**	**FP**
Dataset 4/Mfcc	0.3016	0.1969	0.2833	0.5849	0.8177	76	1503	310	42

Table 8.2 Performance values of machine learning algorithms using dataset 16

Dataset/Alg./Feat.	**F1**	**Recall**	**MCC**	**Bal. Acc.**	**Tot. Acc.**	**TP**	**TN**	**FN**	**FP**
Dataset 16/RF/Mfcc	0.4348	0.3472	0.3859	0.6538	**0.8764**	25	436	47	18

Table 8.2 Performance values of machine learning algorithms using dataset 16 (continue)

Dataset 16/XG/Mfcc	0.4056	0.4028	0.3121	0.6551	0.8384	28	391	44	63
Dataset 16/KNN/Mfcc	0.3077	0.3611	0.1814	0.6024	0.7776	26	383	46	71
Dataset 16/DT/Mfcc	0.3436	0.3889	0.2273	0.6251	0.7966	28	391	44	63
Dataset 16/RF/PSD	0.4408	**0.9306**	0.3926	0.7836	0.6768	67	289	5	165
Dataset 16/XG/PSD	0.3137	0.3333	0.1979	0.6039	0.8004	24	397	48	57
Dataset 16/KNN/PSD	0.3814	0.5139	0.2660	0.6633	0.7719	37	369	35	85
Dataset 16/DT/PSD	**0.5388**	0.9167	**0.4982**	**0.8405**	0.7852	66	347	6	107
Dataset 16/RF/STF	0.4806	0.8611	0.4227	0.7940	0.7452	62	330	10	124
Dataset 16/XG/STF	0.4766	0.8472	0.4154	0.7882	0.7452	61	331	11	123
Dataset 16/KNN/STF	0.4474	0.7083	0.3590	0.7385	0.7605	51	349	21	105
Dataset 16/DT/STF	0.4925	0.9167	0.4481	0.8152	0.7414	66	324	6	130
Dataset 16/SVM/STF	0	0	-	0.5	0.8631	0	454	72	0
Dataset 16/NN/STF	0	0	-	0.5	0.8631	0	454	72	0
Dataset 16/LR/STF	0	0	-	0.5	0.8631	0	454	72	0

8.2.1.2 URTI Class Binary Classifications

Dataset 26 is a dataset labeled URTI vs. Bronchiolitis. Dataset 27 is a dataset labeled URTI vs. Others (Pneumonia and Bronchiolitis). Dataset 30 is a dataset labeled URTI vs. Pneumonia. Dataset 34 is a dataset labeled URTI vs. Others (Healthy, COPD, Pneumonia and Bronchiolitis) as detailed in Chapter 3. Dataset 39 is a dataset labeled

URTI vs. Healthy as described in Chapter 3. As seen in Table 8.3, Decision Tree algorithm achieved the best results in F1 and Recall criteria by using PSD features in Dataset 26. The XGBoost algorithm achieved the best results in the MCC, Balanced and Total Accuracy criteria using the PSD features in Dataset 39.

Table 8.3 Performance values of machine learning algorithms using binary URTI class datasets

Dataset/Alg./Feat.	**F1**	**Recall**	**MCC**	**Bal. Acc.**	**Tot. Acc.**	**TP**	**TN**	**FN**	**FP**
Dataset 26/RF/Mfcc	0.4250	0.3505	-0.3416	0.3128	0.3285	34	11	63	29
Dataset 26/XG/Mfcc	0.3899	0.3196	-0.4160	0.2723	0.2920	31	9	66	31
Dataset 26/KNN/Mfcc	0.6364	0.5773	0.0021	0.5012	0.5328	56	17	41	23
Dataset 26/DT/Mfcc	0.5000	0.4021	-0.0899	0.4510	0.4307	39	20	58	20
Dataset 26/RF/PSD	0.5389	0.4639	-0.1465	0.4195	0.4380	45	15	52	25
Dataset 26/XG/PSD	0.7045	0.6392	0.1971	0.6071	0.6204	62	23	35	17
Dataset 26/KNN/PSD	0.5276	0.4433	-0.1198	0.4341	0.4380	43	17	54	23
Dataset 26/DT/PSD	**0.8035**	**0.9485**	-0.1250	0.4742	0.6715	92	0	5	40
Dataset 26/RF/STF	0.6587	0.5670	0.1746	0.5960	0.5839	55	25	42	15
Dataset 26/XG/STF	0.6474	0.5773	0.0707	0.5387	0.5547	56	20	41	20
Dataset 26/KNN/STF	0.6171	0.5567	-0.0398	0.4784	0.5109	54	16	43	24
Dataset 26/DT/STF	0.6243	0.5567	0.0061	0.5034	0.5255	54	18	43	22
Dataset 27/RF/Mfcc	0.1085	0.0722	-0.1888	0.4328	0.4725	7	96	90	25
Dataset 27/XG/Mfcc	0.2222	0.1649	-0.1103	0.4544	0.4862	16	90	81	31

Table 8.3 Performance values of machine learning algorithms using binary URTI class datasets (continue)

Dataset 27/KNN/Mfcc	0.5079	0.4948	0.1320	0.5656	0.5734	48	77	49	44
Dataset 27/DT/Mfcc	0.3214	0.2784	-0.0904	0.4574	0.4771	27	77	70	44
Dataset 27/RF/PSD	0.3099	0.2268	0.0451	0.5184	0.5505	22	98	75	23
Dataset 27/XG/PSD	0.5204	0.5258	0.1288	0.5645	0.5688	51	73	46	48
Dataset 27/KNN/PSD	0.1970	0.1340	-0.0647	0.4761	0.5138	13	99	84	22
Dataset 27/DT/PSD	0	0	-	0.5	0.5550	0	121	97	0
Dataset 27/RF/STF	0.2414	0.1443	0.1815	0.5515	0.5963	14	116	83	5
Dataset 27/XG/STF	0.3636	0.2474	0.2119	0.5783	0.6147	24	110	73	11
Dataset 27/KNN/STF	0.3750	0.2784	0.1366	0.5565	0.5872	27	101	70	20
Dataset 27/DT/STF	0.4557	0.3711	0.1821	0.5823	0.6055	36	96	61	25
Dataset 30/RF/Mfcc	0.7172	0.7320	0.3634	0.6808	0.6854	71	51	26	30
Dataset 30/XG/Mfcc	0.7944	0.8763	0.5049	0.7406	0.7528	85	49	12	32
Dataset 30/KNN/Mfcc	0.7418	0.8144	0.3738	0.6788	0.6910	79	44	18	37
Dataset 30/DT/Mfcc	0.7016	0.6907	0.3565	0.6787	0.6798	67	54	30	27
Dataset 30/RF/PSD	0.4167	0.3093	0.1123	0.5497	0.5281	30	64	67	17
Dataset 30/XG/PSD	0.7215	0.8144	0.3041	0.6418	0.6573	79	38	18	43
Dataset 30/KNN/PSD	0.3309	0.2371	0.0030	0.5013	0.4775	23	62	74	19

Table 8.3 Performance values of machine learning algorithms using binary URTI class datasets (continue)

Dataset 30/DT/PSD	0	0	-	0.5	0.4551	0	81	97	0
Dataset 30/RF/STF	0.3750	0.2474	0.2114	0.5805	0.5506	24	74	73	7
Dataset 30/XG/STF	0.4697	0.3196	0.3385	0.6351	0.6067	31	77	66	4
Dataset 30/KNN/STF	0.5034	0.3814	0.2448	0.6105	0.5899	37	68	60	13
Dataset 30/DT/STF	0.4713	0.3814	0.1027	0.5487	0.5337	37	58	60	23
Dataset 34/RF/Mfcc	0.4397	0.3370	0.2109	0.5902	0.6184	31	97	61	18
Dataset 34/XG/Mfcc	0.4509	0.4239	0.0598	0.5293	0.5411	39	73	53	42
Dataset 34/KNN/Mfcc	0.5780	0.5435	0.2789	0.6370	0.6473	50	84	42	31
Dataset 34/DT/Mfcc	0.5859	0.6304	0.2118	0.6065	0.6039	58	67	34	48
Dataset 34/RF/PSD	0.3824	0.2826	0.1531	0.5630	0.5942	26	97	66	18
Dataset 34/XG/PSD	0.6590	0.6196	0.4183	0.7054	0.7150	57	91	35	24
Dataset 34/KNN/PSD	0.3053	0.2174	0.0663	0.5261	0.5604	20	96	72	19
Dataset 34/DT/PSD	0.3576	0.2935	0.0168	0.5076	0.5314	27	83	65	32
Dataset 34/RF/STF	0.0404	0.0217	-0.0598	0.4891	0.5411	2	110	90	5
Dataset 34/XG/STF	0.0392	0.0217	-0.1108	0.4761	0.5266	2	107	90	8
Dataset 34/KNN/STF	0.0566	0.0326	-0.1247	0.4685	0.5169	3	104	89	11
Dataset 34/DT/STF	0.0357	0.0217	-0.2267	0.4326	0.4783	2	97	90	18

Table 8.3 Performance values of machine learning algorithms using binary URTI class datasets (continue)

Dataset 39/RF/Mfcc	0.4806	0.3370	0.3011	0.6268	0.5915	31	66	61	6
Dataset 39/XG/Mfcc	0.5000	0.4239	0.0780	0.5383	0.5244	39	47	53	25
Dataset 39/KNN/Mfcc	0.6289	0.5435	0.3103	0.6537	0.6402	50	55	42	17
Dataset 39/DT/Mfcc	0.6409	0.6304	0.1991	0.5999	0.6037	58	41	34	31
Dataset 39/RF/PSD	0.4333	0.2826	0.3361	0.6274	0.5854	26	70	66	2
Dataset 39/XG/PSD	0.7451	0.6196	**0.5791**	**0.7820**	**0.7622**	57	68	35	4
Dataset 39/KNN/PSD	0.3969	0.2826	0.1190	0.5510	0.5183	26	59	66	13
Dataset 39/DT/PSD	0.4154	0.2935	0.1655	0.5704	0.5366	27	61	65	11
Dataset 39/RF/STF	0.0417	0.0217	-0.0194	0.4970	0.4390	2	70	90	2
Dataset 39/XG/STF	0.0412	0.0217	-0.0575	0.4900	0.4329	2	69	90	3
Dataset 39/KNN/STF	0.0600	0.0326	-0.0849	0.4816	0.4268	3	67	89	5
Dataset 39/DT/STF	0.0412	0.0217	-0.0575	0.4900	0.4329	2	69	90	3

8.2.1.3 COPD Class Binary Classifications

Dataset 24 is a dataset labeled as COPD vs. Others (URTI, Pneumonia and Bronchiolitis). Since almost 100% accuracy was achieved in the classification of this dataset, Random Forest and K-NN classifiers were tested with only the MFCC features. As seen in Table 8.4, the Random Forest algorithm achieved the best results in all criteria.

Table 8.4 Performance values of machine learning algorithms using binary COPD class datasets

Dataset/Alg./Feat.	**F1**	**Recall**	**MCC**	**Bal. Acc.**	**Tot. Acc.**	**TP**	**TN**	**FN**	**FP**
Dataset 24/RF/Mfcc	**0.9955**	**1.0000**	**0.9909**	**0.9954**	**0.9955**	222	216	0	2
Dataset 24/KNN/Mfcc	0.9933	0.9955	0.9864	0.9932	0.9932	221	216	1	2

8.2.1.4 Pneumonia Class Binary Classifications

Dataset 25 is a dataset labeled Pneumonia vs. Others (URTI and Bronchiolitis). Dataset 28 is a dataset labeled as Pneumonia vs. Bronchiolitis. Dataset 33 is a dataset labeled as Pneumonia vs. Others (Healthy and URTI) as detailed in Chapter 3. Dataset 38 is a dataset labeled as Pneumonia vs. Others (Healthy and URTI) as described in Chapter 3. As seen in Table 8.5, using STF features, the XGBoost algorithms achieved the best results in the Recall criterion in Dataset 25, in the F1 criterion in Dataset 28 and in the MCC, Balanced and Total Accuracy criteria in Dataset 38.

Table 8.5 Performance values of machine learning algorithms using binary Pneumonia class datasets

Dataset/Alg./Feat.	**F1**	**Recall**	**MCC**	**Bal. Acc.**	**Tot. Acc.**	**TP**	**TN**	**FN**	**FP**
Dataset 25/RF/Mfcc	0.6522	0.5556	0.5146	0.7340	0.7798	45	125	36	12
Dataset 25/XG/Mfcc	0.6861	0.5802	0.5691	0.7573	0.8028	47	128	34	9
Dataset 25/KNN/Mfcc	0.5038	0.4074	0.3256	0.6417	0.7018	33	120	48	17
Dataset 25/DT/Mfcc	0.5614	0.5926	0.2807	0.6430	0.6560	48	95	33	42
Dataset 25/RF/PSD	0.2564	0.1852	0.0415	0.5160	0.6009	15	116	66	21
Dataset 25/XG/PSD	0.4800	0.3704	0.3229	0.6341	0.7018	30	123	51	14

Table 8.5 Performance values of machine learning algorithms using binary Pneumonia class datasets (continue)

Dataset 25/KNN/PSD	0.4550	0.5926	-0.0059	0.4970	0.4725	48	55	33	82
Dataset 25/DT/PSD	0.5146	0.6543	0.1258	0.5644	0.5413	53	65	28	72
Dataset 25/RF/STF	0.6970	0.8519	0.4860	0.7507	0.7248	69	89	12	48
Dataset 25/XG/STF	0.7075	**0.9259**	0.5103	0.7586	0.7156	75	81	6	56
Dataset 25/KNN/STF	0.5727	0.7778	0.2242	0.6115	0.5688	63	61	18	76
Dataset 25/DT/STF	0.6022	0.6914	0.3227	0.6668	0.6606	56	88	25	49
Dataset 28/RF/Mfcc	0.6494	0.6173	0.0407	0.5211	0.5537	50	17	31	23
Dataset 28/XG/Mfcc	0.6203	0.6049	-0.0930	0.4525	0.5041	49	12	32	28
Dataset 28/KNN/Mfcc	0.6400	0.4938	0.3851	0.6969	0.6281	40	36	41	4
Dataset 28/DT/Mfcc	0.6452	0.6173	0.0167	0.5086	0.5455	50	16	31	24
Dataset 28/RF/PSD	0.4065	0.3086	-0.1150	0.4418	0.3967	25	23	56	17
Dataset 28/XG/PSD	0.5758	0.4691	0.1373	0.5721	0.5372	38	27	43	13
Dataset 28/KNN/PSD	0.7101	0.7407	0.0430	0.5204	0.5950	60	12	21	28
Dataset 28/DT/PSD	0.1887	0.1235	-0.2923	0.3742	0.2893	10	25	71	15
Dataset 28/RF/STF	0.8605	0.9136	0.5323	0.7443	0.8017	74	23	7	17
Dataset 28/XG/STF	**0.8757**	0.9136	0.5953	0.7818	0.8264	74	26	7	14
Dataset 28/KNN/STF	0.7692	0.8642	0.1130	0.5446	0.6529	70	9	11	31

Table 8.5 Performance values of machine learning algorithms using binary Pneumonia class datasets (continue)

Dataset 28/DT/STF	0.8537	0.8642	0.5465	0.7696	0.8017	70	27	11	13
Dataset 33/RF/Mfcc	0.5385	0.4321	0.4263	0.6792	0.7786	35	176	46	14
Dataset 33/XG/Mfcc	0.5606	0.4568	0.4487	0.6916	0.7860	37	176	44	14
Dataset 33/KNN/Mfcc	0.3485	0.2840	0.1600	0.5683	0.6827	23	162	58	28
Dataset 33/DT/Mfcc	0.4853	0.4074	0.3319	0.6458	0.7417	33	168	48	22
Dataset 33/RF/PSD	0.1625	0.1605	-0.1882	0.4066	0.5055	13	124	68	66
Dataset 33/XG/PSD	0.4000	0.3457	0.2025	0.5913	0.6900	28	159	53	31
Dataset 33/KNN/PSD	0.3441	0.3951	0.0102	0.5054	0.5498	32	117	49	73
Dataset 33/DT/PSD	0.2822	0.2840	-0.0265	0.4867	0.5683	23	131	58	59
Dataset 33/RF/STF	0.6818	0.5556	0.6137	0.7620	0.8450	45	184	36	6
Dataset 33/XG/STF	0.8000	0.7160	0.7377	0.8422	0.8930	58	184	23	6
Dataset 33/KNN/STF	0.5466	0.5432	0.3550	0.6769	0.7306	44	154	37	36
Dataset 33/DT/STF	0.6842	0.6420	0.5642	0.7710	0.8229	52	171	29	19
Dataset 38/RF/Mfcc	0.5426	0.4321	0.4182	0.6764	0.7592	35	151	46	13
Dataset 38/XG/Mfcc	0.5692	0.4568	0.4512	0.6918	0.7714	37	152	44	12
Dataset 38/KNN/Mfcc	0.3511	0.2840	0.1393	0.5597	0.6531	23	137	58	27
Dataset 38/DT/Mfcc	0.4962	0.4074	0.3354	0.6458	0.7265	33	145	48	19

Table 8.5 Performance values of machine learning algorithms using binary Pneumonia class datasets (continue)

Dataset 38/RF/PSD	0.1739	0.1481	-0.1406	0.4369	0.5347	12	119	69	45
Dataset 38/XG/PSD	0.4088	0.3457	0.1960	0.5875	0.6694	28	136	53	28
Dataset 38/KNN/PSD	0.3699	0.3951	0.0284	0.5146	0.5551	32	104	49	60
Dataset 38/DT/PSD	0.3108	0.2840	0.0165	0.5078	0.5837	23	120	58	44
Dataset 38/RF/STF	0.7143	0.5556	0.6749	0.7778	0.8531	45	164	36	0
Dataset 38/XG/STF	0.8345	0.7160	**0.7925**	**0.8580**	**0.9061**	58	164	23	0
Dataset 38/KNN/STF	0.6154	0.5432	0.4690	0.7167	0.7755	44	146	37	18
Dataset 38/DT/STF	0.7482	0.6420	0.6700	0.8027	0.8571	52	158	29	6

8.2.1.5 Bronchiolitis Class Binary Classifications

Dataset 29 is a dataset labeled Bronchiolitis vs. Others (URTI and Pneumonia). Dataset 32 is a dataset labeled Bronchiolitis vs. Others (Healthy, URTI and Pneumonia) as detailed in Chapter 3. Dataset 37 is a dataset labeled as Bronchiolitis vs. Others (Healthy, URTI and Pneumonia) as described in Chapter 3. As seen in Table 8.6, the Decision Tree algorithm achieved the best results in Dataset 32 in F1, MCC and Total Accuracy using the PSD features. The XGBoost algorithm achieved the best results in the Recall and Balanced Accuracy criteria in Dataset 29 using the STF features.

Table 8.6 Performance values of machine learning algorithms using binary bronchiolitis class datasets

Dataset/Alg./Feat.	**F1**	**Recall**	**MCC**	**Bal. Acc.**	**Tot. Acc.**	**TP**	**TN**	**FN**	**FP**

Table 8.6 Performance values of machine learning algorithms using binary bronchiolitis class datasets (continue)

Dataset 29/RF/Mfcc	0.0606	0.0750	-0.2087	0.3802	0.5734	3	122	37	56
Dataset 29/XG/Mfcc	0.0594	0.0750	-0.2163	0.3746	0.5642	3	120	37	58
Dataset 29/KNN/Mfcc	0.2586	0.3750	0.0262	0.5162	0.6055	15	117	25	61
Dataset 29/DT/Mfcc	0.1786	0.2500	-0.0809	0.4508	0.5780	10	116	30	62
Dataset 29/RF/PSD	0.2400	0.5250	-0.0920	0.4423	0.3899	21	64	19	114
Dataset 29/XG/PSD	0.3619	0.4750	0.1832	0.6083	0.6927	19	132	21	46
Dataset 29/KNN/PSD	0.1935	0.2250	-0.0200	0.4889	0.6560	9	134	31	44
Dataset 29/DT/PSD	0.4103	0.4000	0.2820	0.6382	0.7890	16	156	24	22
Dataset 29/RF/STF	0.4286	0.4500	0.2931	0.6520	0.7798	18	152	22	26
Dataset 29/XG/STF	0.5000	**0.5750**	0.3743	**0.7060**	0.7890	23	149	17	29
Dataset 29/KNN/STF	0.1944	0.1750	0.0378	0.5173	0.7339	7	153	33	25
Dataset 29/DT/STF	0.4301	0.5000	0.2839	0.6573	0.7569	20	145	20	33
Dataset 32/RF/Mfcc	0	0	-0.1587	0.4206	0.7260	0	212	40	40
Dataset 32/XG/Mfcc	0.0208	0.0250	-0.1688	0.4034	0.6781	1	197	39	55
Dataset 32/KNN/Mfcc	0.1765	0.2250	0.0123	0.5073	0.7123	9	199	31	53
Dataset 32/DT/Mfcc	0.1176	0.1500	-0.0607	0.4639	0.6918	6	196	34	56
Dataset 32/RF/PSD	0.2887	0.3500	0.1556	0.5897	0.7637	14	209	26	43

Table 8.6 Performance values of machine learning algorithms using binary bronchiolitis class datasets (continue)

Dataset 32/XG/PSD	0.1124	0.1250	-0.0456	0.4752	0.7295	5	208	35	44
Dataset 32/KNN/PSD	0.1818	0.1500	0.0853	0.5353	0.8151	6	232	34	20
Dataset 32/DT/PSD	**0.5246**	0.4000	**0.5059**	0.6901	**0.9007**	16	247	24	5
Dataset 32/RF/STF	0.1333	0.0750	0.1777	0.5335	0.8664	3	250	37	2
Dataset 32/XG/STF	0.2083	0.1250	0.2382	0.5565	0.8699	5	249	35	3
Dataset 32/KNN/STF	0.1455	0.1000	0.0878	0.5282	0.8390	4	241	36	11
Dataset 32/DT/STF	0.1176	0.0750	0.0781	0.5216	0.8459	3	244	37	8
Dataset 37/RF/Mfcc	0	0	-0.1600	0.4200	0.7241	0	210	40	40
Dataset 37/XG/Mfcc	0.0208	0.0250	-0.1703	0.4025	0.6759	1	195	39	55
Dataset 37/KNN/Mfcc	0.1765	0.2250	0.0109	0.5065	0.7103	9	197	31	53
Dataset 37/DT/Mfcc	0.1176	0.1500	-0.0622	0.4630	0.6897	6	194	34	56
Dataset 37/RF/PSD	0.2887	0.3500	0.1545	0.5890	0.7621	14	207	26	43
Dataset 37/XG/PSD	0.1124	0.1250	-0.0469	0.4745	0.7276	5	206	35	44
Dataset 37/KNN/PSD	0.1818	0.1500	0.0845	0.5350	0.8138	6	230	34	20
Dataset 37/DT/PSD	0.5246	0.4000	0.5056	0.6900	0.9000	16	245	24	5
Dataset 37/RF/STF	0.2128	0.1250	0.2629	0.5585	0.8724	5	248	35	2
Dataset 37/XG/STF	0.2083	0.1250	0.2379	0.5565	0.8690	5	247	35	3

Table 8.6 Performance values of machine learning algorithms using binary bronchiolitis class datasets (continue)

Dataset 37/KNN/STF	0.1455	0.1000	0.0872	0.5280	0.8379	4	239	36	11
Dataset 37/DT/STF	0.1569	0.1000	0.1300	0.5360	0.8517	4	243	36	7

8.2.2 Binary Classifications with Shapley Methods

In this section, binary classifications were performed for Dataset 31 (COPD vs. Others), 40 (Bronchiolitis vs. Others), 41 (Pneumonia vs. Others), 42 (URTI vs. Others) and 43 (Healthy vs. Others), which are the two-class datasets described in Chapter 3. Feature Set 1, which combines three features (MFCC, PSD and STF) and Feature Sets obtained with Shapley values (Feature Set 2, Feature Sets 3 and 4) described in Chapter 7 were used as input data. These features were not separated as train-test and were used with 5-Fold Cross Validation. In addition, the performance values of Shapley's methods (Methods 2, 3, 4 and 5), which used the Decision Tree algorithm and were described in Section 7, were evaluated comparatively.

8.2.2.1 Healthy and The Others Classifications

The results seen in Table 8.7 show the performance values of 5-fold cross validation for Dataset 43 according to different success criteria of the most effective features created according to Feature Set 1 and Shapley value-based selected features. In the healthy vs unhealthy classification made for Dataset 43, we obtained the best results by Method 4 (using Feature Sets 4) with the values of 0.8442 in Total Accuracy criterion. It is seen that the performance of the features with the highest 7 Shapley values in Feature Sets 4 gives the most successful results in all criteria. Compared to Method 1 (using Feature Set 1) for Dataset 43, we have increased the success in all criteria with the Shapley values, and we have successfully done dimension reduction operations by removing the ineffective features.

Table 8.7 5-fold cross-validation performance of Decision Tree algorithm for dataset 43

Features	MCC	Total Accuracy	F1
Feature Set 1	0.3623	0.8234	0.6702
Feature Set 2 (18 Features)	0.3421	0.7951	0.6577
Feature Sets 3 (24 Features)	0.3832	0.8203	0.6770
Feature Sets 4 (7 Features)	**0.4487**	**0.8442**	**0.7128**
Feature Sets 5 (24 Features)	0.4164	0.8070	0.6916

8.2.2.2 URTI and The Others Classifications

The results seen in Table 8.8 show the performance values of 5-fold cross validation for Dataset 42, according to different success criteria of the most effective features according to PSD, Feature Set 1 and Shapley value-based selected features. In the URTI vs others classification made for Dataset 42, we obtained the best results by Method 2 (using Feature Set 2) with the values of 0.8511 in Total Accuracy criterion. In Feature Set 2, it is seen that the performance of the features with the highest 12 Shapley values is the most successful in all criteria. For Dataset 42, we have increased the performance of Method 1 (using Feature Set 1) in all criteria with the Shapley values and we have successfully done dimension reduction operations by removing the ineffective features.

Table 8.8 5-fold cross-validation performance of Decision Tree algorithm for dataset 42

Features	MCC	Total Accuracy	F1
PSD (43) Feature	0.2541	0.6815	0.5633
Feature Set 1	0.1541	0.8040	0.5526

Table 8.8 5-fold cross-validation performance of Decision Tree algorithm for dataset 42 (continue)

Feature Set 2 (12 Features)	**0.3164**	**0.8511**	**0.6371**
Feature Sets 3 (9 Features)	0.2342	0.8178	0.5999
Feature Sets 4 (12 Features)	0.2497	0.8241	0.6057
Feature Sets 5 (27 Features)	0.2234	0.8222	0.6005

8.2.2.3 COPD and The Others Classifications

The results seen in Table 8.9 show the performance values of 5-fold cross validation for Dataset 31, according to different success criteria of the most effective features according to Feature Set 1 and Shapley value-based selected features. In the COPD vs others classification made for Dataset 31, we obtained the best results by Method 3 (using Feature Sets 3) with the values of 0.9931 in Total Accuracy criterion. It is seen that the performance of the features with the highest 4 Shapley values in Feature Sets 3 gives the most successful results in all criteria. Compared to Method 1 (using Feature Set 1) for Dataset 31, we have increased the success in all criteria with the Shapley values, and we have successfully done dimension reduction operations by removing the ineffective features.

Table 8.9 5-fold cross-validation performance of Decision Tree algorithm for dataset 31

Features	**MCC**	**Total Accuracy**	**F1**
Feature Set 1	0.9540	0.9768	0.9767
Feature Set 2 (12 Features)	0.9726	0.9862	0.9861
Feature Sets 3 (4 Features)	**0.9862**	**0.9931**	**0.9931**
Feature Sets 4 (7 Features)	0.9686	0.9843	0.9842

Table 8.9 5-fold cross-validation performance of Decision Tree algorithm for dataset 31 (continue)

Feature Sets 5 (11 Features)	0.9677	0.9837	0.9836

8.2.2.4 Pneumonia and The Others Classifications

The results given in Table 8.10 show the performance values of 5-fold cross validation for Dataset 41, according to different success criteria of the most effective features according to MFCC, Feature Set 1 and Shapley value-based selected features. In the Pneumonia vs others classification made for Dataset 41, we obtained the best results by Method 5 (using Feature Sets 5) with the values of 0.8618 and 0.7599 in Total Accuracy and F1 criterion, respectively. In Feature Sets 5, it is seen that the performance of the features with the highest 15 Shapley values gives the most successful results in Total Accuracy and F1 criterion. We obtained the best results by Method 4 (using Feature Sets 4) with the values of 0.5557 in MCC criterion. In Feature Sets 4, it is seen that the performance of the features with the highest 28 Shapley values gives the most successful results in MCC criterion. For Dataset 41, we have increased the performance of Method 1 (using Feature Set 1) in all criteria with Shapley values and we have successfully done dimension reduction operations by removing the ineffective features.

Table 8.10 5-fold cross-validation performance of Decision Tree algorithm for dataset 41

Features	**MCC**	**Total Accuracy**	**F1**
MFCC (14) Feature	0.5172	0.8493	0.7489
Feature Set 1	0.4616	0.8285	0.7125
Feature Set 2 (13 Features)	0.5285	0.8505	0.7544
Feature Sets 3 (10 Features)	0.4696	0.8260	0.7139
Feature Sets 4 (28 Features)	**0.5557**	0.8524	0.7568

Table 8.10 5-fold cross-validation performance of Decision Tree algorithm for dataset 41 (continue)

Feature Sets 5 (15 Features)	0.5493	**0.8618**	**0.7599**

8.2.2.5 Bronchiolitis and The Others Classifications

The results seen in Table 8.11 show the performance values of 5-fold cross validation for Dataset 40, according to different success criteria of the most effective features according to PSD, Feature Set 1 and Shapley value-based selected features. In the Bronchiolitis vs others classification, we made for Dataset 40, we obtained the best results by Method 4 (using Feature Sets 4) with the values of 0.8851 in Total Accuracy criterion. In Feature Sets 4, it is seen that the performance of the features with the highest 23 Shapley values gives the most successful results in all criteria. For Dataset 40, compared to Method 1 (using Feature Set 1), we have increased the success in all criteria with the Shapley values and we have successfully done dimension reduction operations by removing the ineffective features.

Table 8.11 5-fold cross-validation performance of Decision Tree algorithm for dataset 40

Features	**MCC**	**Total Accuracy**	**F1**
PSD (43) Feature	0.1729	0.7205	0.5438
Feature Set 1	0.3189	0.8028	0.6221
Feature Set 2 (6 Features)	0.1952	0.8178	0.5751
Feature Sets 3 (21 Features)	0.2524	0.8643	0.6041
Feature Sets 4 (23 Features)	**0.3199**	**0.8851**	**0.6417**
Feature Sets 5 (24 Features)	0.3059	0.8681	0.6294

8.3 Ensemble Classifications

In these classifiers, ensemble systems are designed in which the decisions of 5 binary classifiers made using Shapley methods are evaluated jointly (Kababulut et al., 2023). In these systems, we can think of each binary classifier as doctors specialized in their field (Kababulut et al., 2023). Another doctor then makes a final decision based on their diagnosis (Kababulut et al., 2023). In other words, the results of Decision Tree algorithms (DTAs) that estimate 5 different lung diseases become the input data of one DTA that predicts 5 classes. Ensemble Systems consisting of classifications made with Methods 3 and 4 are shown in Figures 8.1 and 8.2.

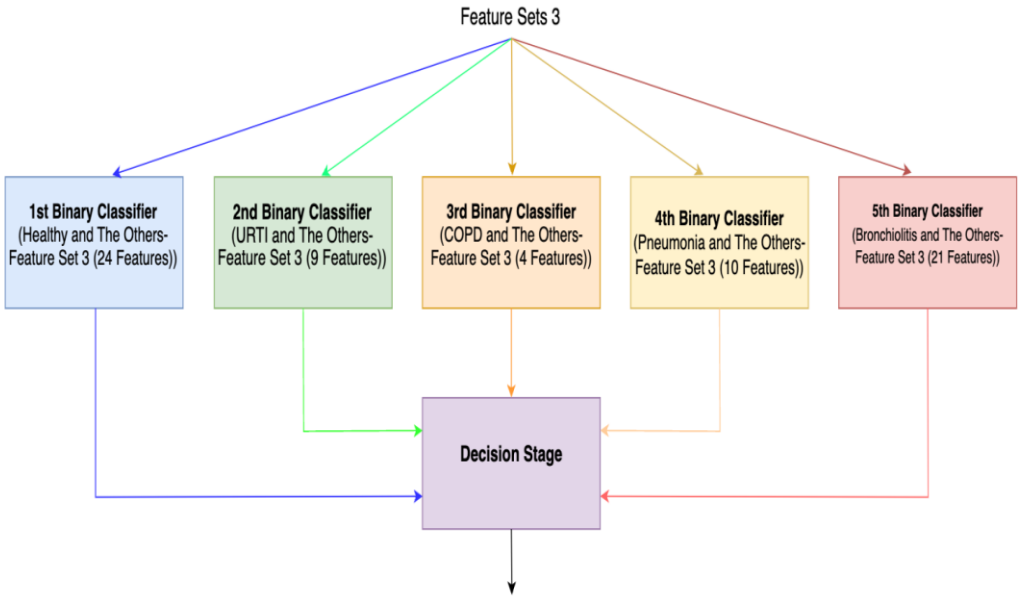


Figure 8.1 Ensemble System (Method 3) (Kababulut et al., 2023)

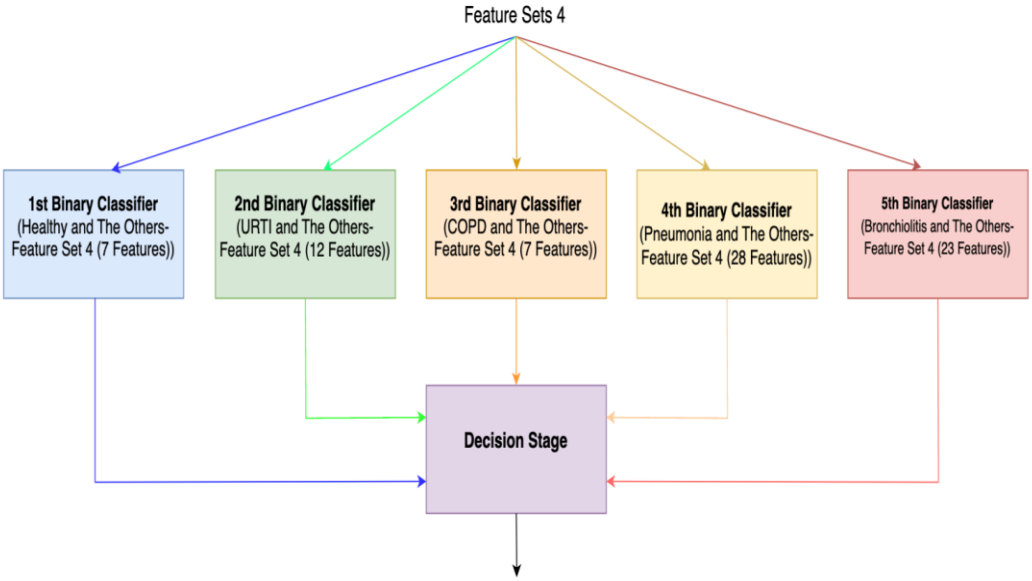


Figure 8.2 Ensemble System (Method 4) (Kababulut et al., 2023)

8.4 Three-Class Classifications

Dataset 11 is a three-class dataset labeled as COPD, Bronchectasy and Others (Healthy, URTI, Pneumonia and Bronchiolitis). Dataset 13 is a three-class dataset labeled as COPD, Bronchectasy and Others (URTI, Pneumonia and Bronchiolitis). Dataset 17 is a dataset labeled as COPD, Bronchectasy and Others (Healthy, URTI, Pneumonia and Bronchiolitis) where records with LPCC values of 0 were deleted. Dataset 18 is a dataset created exactly like Dataset 17 and labeled COPD, Bronchectasy and Others (URTI, Pneumonia and Bronchiolitis). Dataset 35 is a dataset labeled URTI, Healthy and Others (COPD, Pneumonia and Bronchiolitis) as detailed in Chapter 3. As seen in Table 8.12, the Random Forest algorithm achieved the best results in Recall and Balanced Accuracy criteria in Dataset 11 using the PSD features, while it achieved the best results in F1, MCC and Total Accuracy criteria in Dataset 17 using the MFCC features.

Table 8.12 Performance values of Random Forest algorithm using three-class datasets

Dataset/Feat.	**F1**	**Recall**	**MCC**	**Bal. Acc.**	**Tot. Acc.**	**TP**	**TN**	**FN**	**FP**

Table 8.12 Performance values of Random Forest algorithm using three-class datasets (continue)

Dataset 11/Mfcc	0.7296	0.7097	0.7541	0.7097	0.9546	238	330	0	27
Dataset 11/PSD	0.7136	**0.7373**	0.6225	**0.7373**	0.8403	186	314	52	43
Dataset 11/STF	0.3162	0.3259	-0.05	0.3259	0.4840	83	205	155	152
Dataset 13/Mfcc	0.7358	0.7174	0.7454	0.7174	0.9425	238	221	0	28
Dataset 17/Mfcc	**0.7419**	0.7128	**0.7735**	0.7128	**0.9753**	221	292	1	12
Dataset 17/PSD	0.6018	0.6032	0.5409	0.6032	0.8916	182	287	40	17
Dataset 17/STF	0.3177	0.3163	-0.052	0.3163	0.4772	79	172	143	132
Dataset 18/Mfcc	0.7395	0.7128	0.7706	0.7128	0.9714	222	219	0	13
Dataset 18/PSD	0.5983	0.6065	0.5340	0.6065	0.8811	186	214	36	18
Dataset 18/STF	0.2632	0.2608	-0.172	0.2608	0.3789	79	93	143	139
Dataset 35/Mfcc	0.3146	0.3637	-0.0827	0.3637	0.3865	19	61	73	54
Dataset 35/PSD	0.3674	0.4333	0.0884	0.4333	0.4686	25	72	67	43
Dataset 35/STF	0.3883	0.4662	0.2098	0.4662	0.4300	2	87	90	28

8.5 Four-Class Classifications

Dataset 19 is a four-class dataset labeled as COPD, URTI, Pneumonia and Bronchiolitis. As can be seen in Table 8.13, the Random Forest algorithm obtained better results in all criteria for this dataset using the MFCC features compared to other features.

Table 8.13 Performance values of Random Forest algorithm using dataset 19

Dataset/Feat.	**F1**	**Recall**	**MCC**	**Bal. Acc.**	**Tot. Acc.**	**TP**	**TN**	**FN**	**FP**
Dataset19/Mfcc	**0.4988**	**0.4877**	**0.3980**	**0.4877**	**0.6773**	222	76	0	142
Dataset19/PSD	0.4206	0.4170	0.2284	0.4170	0.5819	185	71	37	147
Dataset19/STF	0.1685	0.1854	-0.3326	0.1854	0.2386	79	26	143	192

8.6 Five-Class Classifications

Dataset 20 is a five-class dataset labeled as Healthy, URTI, Bronchectasy, Pneumonia and Bronchiolitis. Dataset 36 is a five-class dataset labeled as Healthy, URTI, COPD, Pneumonia and Bronchiolitis. As can be seen in Table 8.14, the Decision Tree algorithm achieved the best results in Dataset 36 for all criteria using Feature Set 2, which is described in detail in Chapter 7.

Table 8.14 Performance values of machine learning algorithms using five-class datasets

Dataset/Feat.	**F1**	**Recall**	**MCC**	**Bal. Acc.**	**Tot. Acc.**	**TP**	**TN**	**FN**	**FP**
Dataset20/RF/Mfcc	0.2946	0.2871	0.0269	0.2871	0.3289	41	59	31	173
Dataset20/RF/PSD	0.2328	0.2899	0.0170	0.2899	0.3322	69	32	3	200
Dataset20/RF/STF	0.2304	0.3007	0.0625	0.3007	0.3750	65	49	7	183
Dataset36/DT/Mfcc	0.4894	0.4957	0.3843	0.4957	0.6406	44	284	28	156
Dataset36/DT/PSD	0.4520	0.4804	0.3174	0.4804	0.5781	56	240	16	200
Dataset36/DT/STF	0.1983	0.2766	-0.0887	0.2766	0.2930	66	84	6	356

Table 8.14 Performance values of machine learning algorithms using five-class datasets (continue)

Dataset36/DT/Feature Set 1	0.5137	0.5245	0.4256	0.5245	0.6641	40	300	32	140
Dataset36/DT/Feature Set 2	**0.5706**	**0.6078**	**0.5014**	**0.6078**	**0.6934**	54	301	18	139

8.7 Six-Class Classifications

Dataset 21 is a six-class dataset labeled as Healthy, URTI, COPD, Bronchectasy, Pneumonia and Bronchiolitis. Random Forest and K-NN algorithms were tested using only the MFCC features due to low classification performance. As seen in Table 8.15, the K-NN algorithm was the most successful classifier for this dataset in all criteria.

Table 8.15 Performance values of machine learning algorithms using dataset 21

Alg./Feat.	**F1**	**Recall**	**MCC**	**Bal. Acc.**	**Tot. Acc.**	**TP**	**TN**	**FN**	**FP**
Random Forest /Mfcc	0.4104	0.3995	0.3426	0.3995	0.6065	39	280	33	174
K-NN /Mfcc	**0.4650**	**0.4483**	**0.3807**	**0.4483**	**0.6103**	34	287	38	167

8.8 Simulations Results

Since we know that classification success will decrease as the number of classes increases, we primarily focused on binary classifications. In binary classifications, we measured the performance of different classifiers and features with and without Shapley methods. When we looked at the results in the classification of Dataset 4 (Healthy vs. Unhealthy) among the healthy class binary classifications without Shapley methods, we observed that the MFCC segmentation method was not successful. In Dataset 16 (Healthy vs. Unhealthy), we saw that the Decision Tree algorithm classified the Healthy class with 91.66% accuracy using the PSD features. However, this classifier distinguished the Unhealthy class, which includes the other 5 classes, with a lower accuracy of 76.43%. When the Random Forest classifier used the

PSD features, it classified the Healthy class with a higher accuracy of 93.06%, while when it used the MFCC features, it separated the Unhealthy class with a higher accuracy of 96.04%. When we compared the healthy class classifications with each other as methods (Tables 8.1, 8.2 and 8.7), we could see that Method 4 was more successful in F1 (0.7128) criterion, while the Decision Tree algorithm using Dataset 16 and PSD features was more successful than Method 4 using Dataset 43 (Healthy vs. Others) in the MCC (0.4982) criterion. Random Forest algorithm using Dataset 16 and MFCC features was more successful than Method 4 in the Total Accuracy (0.8764) criterion. It should also be noted that Dataset 43 used by the Decision Tree algorithm in Method 4 differs from Dataset 16 in that the Others class don't include the Bronchectasy class.

In the URTI class binary classifications without Shapley methods, the XGBoost algorithm classifying Dataset 39 (URTI vs. Healthy) did not separate the URTI class very well (61.96%) using the PSD features. However, this classifier separated the Healthy class with a high performance of 94.44%. When classifying Dataset 26 (URTI vs. Bronchiolitis) using the PSD features, the Decision Tree algorithm didn't correctly classify any Bronchiolitis records but classified URTI records with a high accuracy of 94.85%. When we compared the URTI class classifications in terms of methods (Tables 8.3 and 8.8), we could see that Method 2 was more successful in the Total Accuracy criterion (0.8511), while the XGBoost algorithm using Dataset 39 and PSD features was more successful than Method 2 using Dataset 42 (URTI vs. Others) in the MCC (0.5791) criterion. Decision Tree algorithm using Dataset 26 and PSD features was more successful than Method 2 in the F1 (0.8035) criterion. It is also worth noting that Dataset 42 used by the Decision Tree algorithm in Method 2, differs from Datasets 39 and 26 in that the Others class contains 4 classes instead of a single class.

In the COPD class binary classifications without Shapley methods, we saw that the Random Forest algorithm classifying Dataset 24 (COPD vs. Others) separated both classes with close to 100% accuracy in all metrics using the MFCC features. When we compared the COPD class classifications as methods (Tables 8.4 and 8.9), we saw that the Random Forest algorithm using Dataset 24 was more successful than Method 3

using Dataset 31 (COPD vs. Others) in all metrics. However, we should mention that Dataset 31 (COPD vs. Others) differs from Dataset 24 in that the Others class contains 4 classes instead of three classes.

In the Pneumonia class binary classifications without Shapley methods, the XGBoost algorithm classified Dataset 38 (Pneumonia vs. Others) with 71.60% accuracy using the STF features and 100% accuracy for the Others class. Using the STF features, the XGBoost algorithm separated the Pneumonia class with 91.36% accuracy in Dataset 28 (Pneumonia vs. Bronchiolitis), while it separated the Pneumonia class with 92.59% accuracy in Dataset 25 (Pneumonia vs. Others). When we compared the Pneumonia class classifications as methods (Tables 8.5 and 8.10), we saw that the XGBoost algorithm using Dataset 38 and STF features was more successful in the MCC (0.7925) and Total Accuracy (0.9061) criteria, while the XGBoost algorithm using Dataset 28 and STF features was more successful than Methods 4 and 5 using Dataset 41 (Pneumonia vs. Others) in the F1 (0.8757) criteria. However, it should be noted that Dataset 41 differs from Datasets 38 and 28 in that the Others class contains 4 classes instead of one or two classes.

In the Bronchiolitis class binary classifications without Shapley methods, the Decision Tree algorithm classifying Dataset 32 (Bronchiolitis vs. Others) using the PSD features separated the Bronchiolitis class with a low accuracy of 40%, while it separated the Others class with a high accuracy of 98.02%. Using the STF features, the XGBoost algorithm separated the Bronchiolitis class with 57.5% accuracy in Dataset 29 (Bronchiolitis vs. Others) classification, while it separated the Others class with 83.71% accuracy. When we compared the Bronchiolitis class classifications in terms of methods (Tables 8.6 and 8.11), Method 4 using Dataset 40 (Bronchiolitis vs. Others) was more successful in the F1 (0.6417) criterion, while the Decision Tree algorithm using Dataset 32 and PSD features was more successful in the MCC (0.5059) and Total Accuracy (0.9007) criteria. However, it should be noted that Dataset 40 differs from Dataset 32 in that the Others class contains 4 classes instead of three.

When we examined the three-class predictions, the Random Forest algorithm classifying Dataset 17 (COPD, Bronchectasy and Others) separated the COPD class with 97.53% accuracy, the Bronchectasy class with 14.29% accuracy and the Others class with 100% accuracy using the MFCC features. The Random Forest algorithm classifying Dataset 11 (COPD, Bronchectasy and Others) separated COPD class with 78.15% accuracy, Bronchectasy class with 51.61% accuracy and Others class with 91.41% accuracy using PSD features. It should be noted that the difference of Dataset 17 compared to Dataset 11 is that records with zero LPCC values were deleted in Dataset 17.

When we examined the four-class classifications, the Random Forest algorithm using Dataset 19 (COPD, URTI, Pneumonia and Bronchiolitis) and the MFCC features classified the COPD class with 100% accuracy, URTI class with 25.77% accuracy, Pneumonia class with 56.79% accuracy and Bronchiolitis class with 12.5% accuracy.

When we examined the five-class classifications, the Decision Tree algorithm using Dataset 36 (Healthy, URTI, COPD, Pneumonia and Bronchiolitis) and Feature Set 2 classified the Healthy class with 75% accuracy, URTI class with 22.68% accuracy, COPD class with 96.85% accuracy, Pneumonia class with 49.38% accuracy and Bronchiolitis class with 60% accuracy.

When we examined the six-class classifications, the K-NN algorithm using Dataset 21 (Healthy, URTI, COPD, Bronchectasy, Pneumonia and Bronchiolitis) and the MFCC features classified the Healthy class with 47.22% accuracy, URTI class with 30.93% accuracy, COPD class with 99.55% accuracy, Bronchectasy class with 42.86% accuracy, Pneumonia class with 25.93% accuracy and Bronchiolitis class with 22.50% accuracy.

In our Ensemble systems, the output in the classification phase can include more than one decision. For example, in our Method 4 Ensemble system shown in Figure 8.1, if Classifier 1 made the Healthy decision, Classifier 2 made the URTI decision and the other classifiers made the Others decision, the output of the system will be

both Healthy and URTI. In this case, according to the results in Tables 8.7 to 8.11, the decision of the classifier with the higher precision criterion was considered correct (Kababulut et al., 2023). In other words, we accepted the decision of the doctor who has made more accurate decisions so far (Kababulut et al., 2023). In this way, precision of the Method 4 Ensemble system was 71.58% (Kababulut et al., 2023). Table 8.16 shows that the Method 4 Ensemble system was more successful than Method 2 (a single DTA), which classified 5 classes (Kababulut et al., 2023).

Table 8.16 Comparison of the performances of different methods (Kababulut et al., 2023)

Method	**Precision**
1	0.5425
2	0.5926
3 (Ensemble)	0.7078
4 (Ensemble)	**0.7158**

Looking at the results in Tables 8.7 to 8.11, it is observed that for all binary classifiers, when the features obtained according to Shapley values were used, the performance values increased compared to the use of Feature set 1 (the merged form of three features). Also, when we looked at the 5-class classifications, we can see from Table 8.16 that Method 2 using Shapley values was more successful than Method 1 using Feature Set 1 (Kababulut et al., 2023). In other words, we found that our DTAs using Shapley values for two-class and five-class classification improved their performance.

Looking at the performance of the binary classifiers using Shapley's methods (Tables 8.7 to 8.11), we saw that Method 4 for Healthy and Bronchiolitis classifications were the most successful in all criteria, Method 2 for URTI classification was the most successful in all criteria, Method 3 for COPD classification was the most successful in all criteria, Method 4 for Pneumonia classification was the most successful in the MCC criterion, and Method 5 was the most successful in the F1 and Total Accuracy criteria. Success order of the classifiers according to the Total

Accuracy criterion: Classifier 3 (COPD and The Others), Classifier 5 (Bronchiolitis and The Others), Classifier 4 (Pneumonia and The Others), Classifier 2 (URTI and The Others) and Classifier 1 (Healthy and The Others) while the ranking according to MCC,F1 and Precision criteria: Classifier 3 (COPD and The Others), Classifier 4 (Pneumonia and The Others), Classifier 1 (Healthy and The Others), Classifier 5 (Bronchiolitis and The Others) and Classifier 2 (URTI and The Others).

When we look at the overall performance of the machine learning algorithms used in the classifications, we saw that Random Forest, XGBoost, K-NN and Decision Tree algorithms got the best results in at least one classification. However, we also found that the XGBoost algorithm was more successful in more classifications than other algorithms.

The reason why we use our Decision Tree algorithm instead of high-performance classifiers in Shapley methods is Python's Shapley library (Kababulut et al., 2023). We can get the contributions of Shapley values to classes as seen in Figure 7.1 in Shapley library with only decision tree-based algorithms (Kababulut et al., 2023). With the Decision tree algorithm, we can determine which features are important in the diagnosis of the disease by looking at the tree structure for that patient.

In the Decision Tree algorithm, the selection of features according to their gini impurity values is based on their contribution to the classification performance alone (Kababulut et al., 2023). However, the selection of features according to their Shapley values is based on their shares in the total contribution to the classification performance (Kababulut et al., 2023). In both methods, we observed that although the ranking of the features according to their contribution varies, the majority of the top 10 features used are the same (Kababulut et al., 2023). For example, Common features for Dataset 40 are the 13th, 6th, 68th, 66th, 18th, and 59th features. The 59th feature is Standard Deviation, the 66th feature is Peak to Peak, and the 68th feature is Impulse Factor from the statistical features. The 6th and 13th features are MFCC features. The 18th feature is the 4th PSD feature. As stated in the study (Ali et al., 2012), wheeze from abnormal lung sounds is observed in bronchiolitis patients. In the studies

(Guntupalli et al., 2008; Rao et al., 2019), it is stated that the wheeze sound is seen in the frequency band of 100-2000 Hz. For Dataset 40, the effective frequency ranges of the 6th MFCC feature are 133.33-266.67 Hz and 266.67-400.01 Hz, the effective frequency ranges of the 13th MFCC feature are 266.67-400.01 Hz and the frequency range of the 4th PSD feature is 330.75-441.00 Hz and the frequency range of all of these features coincides with the frequency range reported in (Guntupalli et al., 2008; Rao et al., 2019). The frequency range of abnormal lung sounds seen in lung diseases in other datasets also overlaps the frequency range of the features we used.

When we examine the features in Feature Sets 4 used in Method 4 in detail, we can observe which features have high ensemble contributions. 3 of the 7 features used in the 1st Classifier (H/O) are MFCC, 3 are PSD, and 1 is STF features (Kababulut et al., 2023). 7 of the 12 features used in Classifier 2 (U/O) are MFCC, 4 are STF and 1 is PSD features (Kababulut et al., 2023). 5 of the 7 features used in the 3rd Classifier (C/O) are MFCC and 2 are PSD features (Kababulut et al., 2023). 13 of the 28 features used in the 4th Classifier (P/O) are MFCC, 9 are STF and 6 are PSD features. We found that 11 of the 23 features used in the 5th Classifier (H/O) are MFCC, 7 are STF, and 5 are PSD features (Kababulut et al., 2023). As can be seen from these results, the feature that contributes the most to classification performance in Shapley methods is MFCC (Kababulut et al., 2023). In classifications without Shapley methods, we can more easily observe the contribution of the features alone to the performance. In these classifications, we found that most of the algorithms with high performance values use the STF features. As can be seen, the contribution of a feature individually and its contribution in the ensemble can be different. When the 3 features are merged, the MFCC feature contributes the most, but when the features are used separately, the STF feature contributes the most.

Finally, the processes applied when using Shapley's methods are shown in Figure 8.3 to better explain our works.

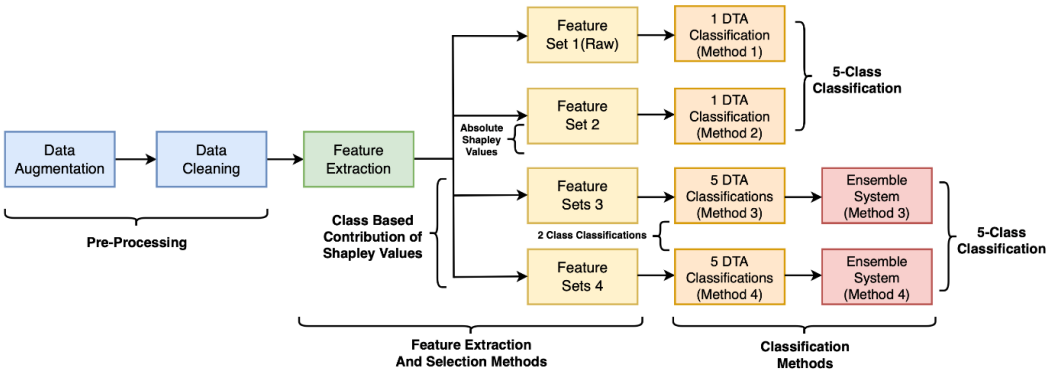


Figure 8.3 Processes of Shapley methods (Kababulut et al., 2023)

CHAPTER NINE
DISCUSSION

The analysis of lung sounds and lung disease diagnosis involves many stages. In addition to data augmentation, data cleaning, feature extraction, feature selection and classification, we also used data visualization in this thesis to evaluate the performance of classifications based on class separabilities.

In this thesis, we used ICBHI 2017 database, which is an open access source widely studied, but the class distribution in this database was imbalanced. Although there were 7 classes with 6 lung diseases and healthy cases, the percentage of lung sound recordings in the COPD class was 86.20% of the total recordings. The database included information on when the respiratory cycles of each recording started and ended. We treated the respiratory cycles of each recording, except for the COPD class, as a separate sound recording and in this way we augmented the data for non-COPD classes. After this process and data cleaning, the percentage of COPD recordings in our dataset was reduced to 45.91% while the percentage of recordings in other classes became 54.09%. After that, we used five different data sampling and augmentation methods to make our imbalanced dataset more balanced. We applied downsampling and upsampling methods to the minority and majority classes of Dataset 1, one of our authentic datasets. While we could not achieve significant success with upsampling methods, we achieved success in all performance criteria with RUS and Tomek methods from down sampling methods. When Dataset 1 was balanced with ADASYN, which is a synthetic data generation method, a significant performance increase was achieved in the other 3 criteria except F1 and Total Accuracy with the Random Forest algorithm. When we balanced the distribution of Dataset 1 by adding noise, the desired success was not achieved. We created new synthetic datasets by augmenting synthetic data from different datasets with the GAN method. When we gave these datasets as input data to the Random Forest algorithm, we observed partial improvements compared to the performance of the authentic datasets. However, the performance of the Random Forest algorithm using Dataset 12, the synthetic dataset obtained from the original Dataset 11, showed significant increases in the other four criteria except Total

Accuracy compared to Dataset 11. From this result, we think that the performance of the algorithms can be improved in all criteria when synthetic data is generated with GAN with appropriate number and technique in appropriate classification.

We used three different methods to visualize the data distribution in two-dimensional representations. The joint interpretation of U-Matrix and Som-Hit maps of the features extracted from the lung sounds mentioned in the third chapter is an approach that we have not encountered in the literature before. Therefore, we think that this approach will be a contribution to the literature. From the U-Matrix and Som-Hit maps we obtained with different classification combinations for all three features, we observed that except for the COPD class, the other 5 classes were mixed up with each other. We also reached the same conclusion from the t-SNE and Dbscan methods.

The feature sets used were obtained as follows. Feature set 1 is the merging of 3 different features (MFCC, PSD, STF) extracted from the lung sounds. Feature set 2 is obtained according to the ranking of Shapley absolute values traditionally used in the literature to measure the ensemble contribution of the 68 features in Feature set 1 to the Decision Tree algorithm. The utilization of the other three feature sets are methods that have not been encountered before in lung sound classification applications and will make a humble contribution to the literature. Feature sets 3 and 4 were obtained according to the class contributions of the features in quintuple and binary classifications, respectively. The use of relative Shapley values (the percentage value of the class contributions of the features in the quintuple classification) in Feature set 5 has not been encountered not only in lung sound classification problems but also in any study in the literature.

We examined the success of classifications made with and without Shapley methods. As expected, the most successful results in these classifications belonged to binary classifications. While almost 100% success was achieved in the COPD vs others classifications, successful results (Total Accuray = 0.9061, Balanced Accuracy = 0.8580, MCC = 0.7925) were also obtained in the Pneumonia vs others classification (Dataset 38). But the desired results could not be obtained in other classifications. It is

clear that the most important reason for this is our imbalanced data set. It is known that machine learning algorithms do not achieve good results with test data when they are not trained with enough data. Meanwhile, it is seen that the classification results of Shapley methods are slightly lower than the results of other classifications. We can say that the reason for this is that the others class in the datasets used in Shapley methods contains more classes. Our main purpose in using Shapley methods was to increase the performance of our own algorithms. Looking at the results in Tables 8.7 to 8.11, we can see that Shapley methods increase the performance of our Decision Tree algorithms in all criteria compared to the use of raw features (Feature Set 1).

Most studies on the ICBHI 2017 database and lung sound/disease classification extract features as 2D images and use image processing techniques. The difference between our work and these studies is that we extracted features from one-dimensional lung sound data. We found only one study (Perna & Tagarelli, 2019) in the literature that used this database and classified with one-dimensional features, in which MFCC features were used as input data to classify healthy-unhealthy pathology with the LSTM algorithm with an accuracy score of 99%. Our DTA, on the other hand, achieved an accuracy score of 84.42% in healthy-unhealthy pathology classification with the same database. In our thesis, instead of using algorithms such as LSTM, RNN and CNN in order to achieve high accuracy in lung disease diagnosis, we preferred to use DTA, which allows feature selection with class-based Shapley values. Additionally, in this study, unlike other studies (Ntalampiras, 2020), we used data sets independent of the type of stethoscope and the place where lung sounds were listened to. In fact, this information was available in our database, and we could further increase our performance with the data sets created by paying attention to this information. However, we wanted to design a system more suitable for real life that could be used later for different data sets.

The main reason why we used DTA in Shapley methods was that decision tree-based algorithms had to be used in Python's Shapley library to obtain the Shapley values of the features for all classes and the graph showing these values. With DTA, we can obtain not only class-based Shapley values but also information about which

features are important for disease estimation from the traceable tree structure. We observed that the important features seen in the Shapley value graphs and the important features seen in the tree structure of DTA are common. It was important that the frequency bands of these features were the same as the frequency bands of abnormal lung sounds that help diagnose lung diseases. This result shows that the selection of features to help diagnose lung disease is correct.

We also developed ensemble systems that combine 5 binary class estimations using Methods 3 and 4 of Shapley's methods into a single quintuple class estimation. Our ensemble system using Method 4 outperformed a single DTA using Method 2, which also estimated quintuple classes. Also, the Shapley methods (Methods 2, 3 and 4) were more successful in these quintuple classifications than Method 1 that uses only raw features. We compared our same ensemble system with another ensemble system (Ntalampiras, 2020) that utilized the ICBHI 2017 database and performed binary classification as abnormal-normal lung sounds. Although our system performed 5-class classification, it achieved an accuracy score of 69.67%, which is better than the 66.70% accuracy score of the classifier in this study. Based on these results, we can list the contributions of Shapley values to our system as follows. First, it gave information about which features are important in lung disease diagnosis. Second, after the feature selection process, it increased the performance of our algorithms in all criteria and reduced the processing load and time of the algorithms by dimension reduction.

CHAPTER TEN
CONCLUSION

Our aim in this thesis was to develop new methods to assist doctors and researchers in the diagnosis of lung diseases using analysis of lung sounds. The class distribution of the database we used was unbalanced. Upon comparing the methods employed for imbalanced datasets with the algorithms that directly utilized imbalanced datasets, we observed that these methods enhanced the performance values of the algorithms. Additionally, we gained more detailed insights into our datasets through the visualization methods employed to examine the data distribution. We think that the use of U-Matrix and Som-Hit maps, among these methods, in lung sound analysis will be a contribution to the literature. Additionally, we conducted feature selection operations using the Shapley values of the features, resulting in the creation of five different feature sets. Three of these sets were generated based on the contribution of features to the separation of classes by the Decision Tree algorithm (DTA). We believe that utilizing Shapley values in this manner constitutes a significant contribution to the literature. Our findings indicate that Shapley methods using these feature sets, enhanced all performance metrics of classifiers across the board compared to Method 1, which involves using all raw features (Feature Set 1). Finally, we devised ensemble systems to consolidate binary classifications into a unified five-class classification. Upon comparison with a singular DTA performing five-class classification and another ensemble system in the existing literature, our ensemble classifier employing Method 4 demonstrated more successful outcomes.

While we do not claim that our study will provide great convenience to doctors in the diagnosis of lung disease, we think that the explainable decision provided by our ensemble system will help researchers in this regard. In our next study, we plan to use different features and techniques to achieve higher classification rates. Some of the features we plan to use are Wavelet coefficients and Intric Mode Functions (IMF) features. We also plan to use Auto-Encoder Neural Network and Genetic Algorithm as other feature selection methods. In this thesis, we generated synthetic data using Conditional Tabular GAN (CTGAN). In our upcoming study, we aim to enhance the

performance of our classifiers across all criteria by augmenting data through various GAN methods such as CWGAN, CTAB-GAN, and TabFairGAN.

REFERENCES

Abdoli, S., Cardinal, P., & Koerich, A. L. (2019). End-to-End Environmental Sound Classification using a 1D Convolutional Neural Network. Expert Systems with Applications, 136, 252–263. https://doi.org/10.1016/j.eswa.2019.06.040

Akşit, S. (2002). Acute Respiratory Infections. https://www.ttb.org.tr/STED/sted0402/akut.pdf

Ali, S., Plint, A. C., & Klassen, T. P. (2012). Bronchiolitis. In Kendig & Chernick's Disorders of the Respiratory Tract in Children (pp. 443–452). W.B. Saunders.

Amaral, J. L. M., Lopes, A. J., Jansen, J. M., Faria, A. C. D., & Melo, P. L. (2012). Machine learning algorithms and forced oscillation measurements applied to the automatic identification of chronic obstructive pulmonary disease. Computer Methods and Programs in Biomedicine, 105(3), 183–193. https://doi.org/10.1016/j.cmpb.2011.09.009

Aykanat, M., Kılıç, Ö., Kurt, B., & Saryal, S. (2017). Classification of lung sounds using convolutional neural networks. Eurasip Journal on Image and Video Processing, 2017(1). https://doi.org/10.1186/s13640-017-0213-2

Badnjevic, A., Cifrek, M., Koruga, D., & Osmankovic, D. (2015). Neuro-fuzzy classification of asthma and chronic obstructive pulmonary disease. BMC Medical Informatics and Decision Making, 15(3). https://doi.org/10.1186/1472-6947-15-S3-S1

Bahoura, M., & Pelletier, C. (2004). Respiratory sounds classification using cepstral analysis and gaussian mixture models. Annual International Conference of the IEEE Engineering in Medicine and Biology - Proceedings, 26 I, 9–12. https://doi.org/10.1109/iembs.2004.1403077

Bardou, D., Zhang, K., & Ahmad, S. M. (2018). Lung sounds classification using convolutional neural networks. Artificial Intelligence in Medicine, 88, 58–69. https://doi.org/10.1016/j.artmed.2018.04.008

Bieroza, M., Baker, A., & Bridgeman, J. (2012). Exploratory analysis of excitation-emission matrix fluorescence spectra with self-organizing maps-A tutorial. Education for Chemical Engineers, 7(1). https://doi.org/10.1016/j.ece.2011.10.002

Breiman, L. (2001). Random Forests. Machine Learning, 45, 5–32.

Cavallaro, M., Moiz, H., Keeling, M. J., & McCarthy, N. D. (2021a). Contrasting factors associated with COVID-19-related ICU admission and death outcomes in hospitalised patients by means of Shapley values. PLoS Computational Biology, 17(6). https://doi.org/10.1371/journal.pcbi.1009121

Cavallaro, M., Moiz, H., Keeling, M. J., & McCarthy, N. D. (2021b). Contrasting factors associated with COVID-19-related ICU admission and death outcomes in hospitalized patients by means of shapley values. PLoS Computational Biology, 17(6). https://doi.org/10.1371/journal.pcbi.1009121

Çelebi, G. (2017). Sayısal Stetoskop ile Elde Edilen Kalp Sesi (Fonokardiyogram) Sinyallerinin Bölütlenmesi ve Sınıflandırılması. Hacettepe Üniversity.

Champeney, D. C. (1987). A Handbook of Fourier Theorems. Cambridge University Press.

Chatterjee, J., Sharma, G., Sexena, A., Mehra, A., & Gupta, V. (2019). A Robust Automatic Algorithm for Statistical Analysis and Classification of Lung Auscultations. 2019 6th International Conference on Signal Processing and Integrated Networks, SPIN 2019, 313–318. https://doi.org/10.1109/SPIN.2019.8711646

Chen, C. H., Huang, W. T., Tan, T. H., Chang, C. C., & Chang, Y. J. (2015). Using K-nearest neighbor classification to diagnose abnormal lung sounds. Sensors (Switzerland), 15(6), 13132–13158. https://doi.org/10.3390/s150613132

Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 785–794. https://doi.org/10.1145/2939672.2939785

Chicco, D., Tötsch, N., & Jurman, G. (2021). The matthews correlation coefficient (Mcc) is more reliable than balanced accuracy, bookmaker informedness, and markedness in two-class confusion matrix evaluation. BioData Mining, 14. https://doi.org/10.1186/s13040-021-00244-z

Chomboon, K., Chujai, P., Teerarassammee, P., Kerdprasop, K., & Kerdprasop, N. (2015). An Empirical Study of Distance Metrics for k-Nearest Neighbor Algorithm. Proceedings of the 3rd International Conference on Industrial Application Engineering, 280–285. https://doi.org/10.12792/iciae2015.051

Çiftçi, F. (2023). What is COPD? What are the symptoms and treatment methods? How is the COPD test performed? www.medicalpark.com.tr/koah-nedir/hg-1771

Clifford, G. D., Liu, C., Moody, B., Springer, D., Silva, I., Li, Q., & Mark, R. G. (2016). Classification of normal/abnormal heart sound recordings: The PhysioNet/Computing in Cardiology Challenge 2016. Computing in Cardiology, 43, 609–612. https://doi.org/10.22489/cinc.2016.179-154

Cooley, J. W., Lewis, E. ' A. W., & Welch, P. D. (1969). The Finite Fourier Transform. IEEE Transactions on Audio and Electroacoustics, 17(2), 77–85.

Creswell, A., White, T., Dumoulin, V., Arulkumaran, K., Sengupta, B., & Bharath, A. A. (2017). Generative Adversarial Networks: An Overview. IEEE Signal Processing Magazine, 35. https://doi.org/10.1109/MSP.2017.2765202

Dao, V.-L., Nguyen, V.-D., Nguyen, H.-D., & Hoang, V.-P. (2016). Hardware Implementation of MFCC Feature Extraction for Speech Recognition on FPGA. In M. Akagi, T.-T. Nguyen, D.-T. Vu, T.-N. Phung, & V.-N. Huynh (Eds.), Advances in Information and Communication Technology, ICTA 2016 (pp. 248–254). Springer. http://www.springer.com/series/11156

Dwivedi, A. K., Imtiaz, S. A., & Rodriguez-Villegas, E. (2019). Algorithms for automatic analysis and classification of heart sounds-A systematic review. IEEE Access, 7, 8316–8345. https://doi.org/10.1109/ACCESS.2018.2889437

Elhassan, T. A. M., Aljourf, M., Al-Mohanna, F., & Shoukri, M. (2017). Classification of Imbalance Data using Tomek Link (T-Link) Combined with Random Under-sampling (RUS) as a Data Reduction Method. Global Journal of Technology and Optimization, 01(S1). https://doi.org/10.4172/2229-8711.s1111

Engelmann, J., & Lessmann, S. (2021). Conditional Wasserstein GAN-based Oversampling of Tabular Data for Imbalanced Learning. Expert Systems with Applications, 174. https://doi.org/10.1016/j.eswa.2021.114582

Engin, M. A., Aras, S., & Gangal, A. (2022). Extraction of low-dimensional features for single-channel common lung sound classification. Medical and Biological Engineering and Computing, 60(6), 1555–1568. https://doi.org/10.1007/s11517-022-02552-w

Ester, M., Kriegel, H.-P., Sander, J., & Xu, X. (1996, August 2). A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. 2.

International Conference on Knowledge Discovery and Data Mining. www.aaai.org

Florence Nightingale. (2023). What are Upper Respiratory Tract Infections? Symptoms and Treatment Methods. https://www.florence.com.tr/ust-solunum-yolu-enfeksiyonu#:~:text=%C3%9Cst%20solunum%20yolu%20enfeksiyonu%2C%20t%C4%B1pta,enfeksiyonu%2C%20genellikle%20rinovir%C3%BCsler%20sonucu%20olu%C5%9Fur.

Fraiwan, L., Hassanin, O., Fraiwan, M., Khassawneh, B., Ibnian, A. M., & Alkhodari, M. (2021). Automatic identification of respiratory diseases from stethoscopic lung sound signals using ensemble classifiers. Biocybernetics and Biomedical Engineering, 41(1), 1–14. https://doi.org/10.1016/j.bbe.2020.11.003

García-Ordás, M. T., Benítez-Andrades, J. A., García-Rodríguez, I., Benavides, C., & Alaiz-Moretón, H. (2020). Detecting respiratory pathologies using convolutional neural networks and variational autoencoders for unbalancing data. Sensors (Switzerland), 20(4). https://doi.org/10.3390/s20041214

Gavriely, N., & Cugell, D. W. (2019). Breath sounds methodology. CRC Press.

Grønnesby, M., Carlos, J., Solis, A., Holsbø, E., Melbye, H., & Bongo, L. A. (2017). Feature Extraction for Machine Learning Based Crackle Detection in Lung Sounds from a Health Survey. https://uit.no/forskning/forskningsgrupper/sub?sub_id=503778&p_document_id=367276

Güler, H. C., Yildiz, O., Baysal, U., Cinyol, F. B., Köksal, D., Babaoglu, E., & Sarinç Ulasli, S. (2020, November 19). Classification of Abnormal Respiratory Sounds Using Machine Learning Techniques. TIPTEKNO 2020 - Tip Teknolojileri Kongresi - 2020 Medical Technologies Congress, TIPTEKNO 2020. https://doi.org/10.1109/TIPTEKNO50054.2020.9299294

Guntupalli, K. K., Alapat, P. M., Bandi, V. D., & Kushnir, I. (2008). Validation of automatic wheeze detection in patients with obstructed airways and in healthy subjects. Journal of Asthma, 45(10), 903–907. https://doi.org/10.1080/02770900802386008

Gupta, C. N., Palaniappan, R., Swaminathan, S., & Krishnan, S. M. (2007). Neural network classification of homomorphic segmented heart sounds. Applied Soft Computing Journal, 7(1), 286–297. https://doi.org/10.1016/j.asoc.2005.06.006

Gurung, A., Scrafford, C. G., Tielsch, J. M., Levine, O. S., & Checkley, W. (2011). Computerized lung sound analysis as diagnostic aid for the detection of abnormal lung sounds: A systematic review and meta-analysis. Respiratory Medicine, 105(9), 1396–1403. https://doi.org/10.1016/j.rmed.2011.05.007

Habukawa, C., Murakami, K., Endoh, M., Horii, N., & Nagasaka, Y. (2017). Treatment evaluation using lung sound analysis in asthmatic children. Respirology, 22(8), 1564–1569. https://doi.org/10.1111/resp.13109

Haider, N. S., & Behera, A. K. (2022). Computerized lung sound based classification of asthma and chronic obstructive pulmonary disease (COPD). Biocybernetics and Biomedical Engineering, 42(1), 42–59. https://doi.org/10.1016/j.bbe.2021.12.004

Hamming, R. W. (1989). Digital filters (3rd edition). Prentice Hall.

He, H., Bai, Y., Garcia, E. A., & Li, S. (2008). ADASYN: Adaptive synthetic sampling approach for imbalanced learning. 2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence), 1322–1328. https://doi.org/10.1109/IJCNN.2008.4633969

Hearst, M. A., Dumais, S. T., Osuna, E., Platt, J., & Scholkopf, B. (1998). Support vector machines. IEEE Intell. Syst. Their Appl., 13, 18–28.

Heitmann, J., Glangetas, A., Doenz, J., Dervaux, J., Shama, D. M., Garcia, D. H., Benissa, M. R., Cantais, A., Perez, A., Müller, D., Chavdarova, T., Ruchonnet-Metrailler, I., Siebert, J. N., Lacroix, L., Jaggi, M., Gervaix, A., Hartley, M. A., Hugon, F., Fassbind, D., … Ba, I. (2023). DeepBreath—automated detection of respiratory pathology from lung auscultation in 572 pediatric outpatients across 5 countries. Npj Digital Medicine, 6(1). https://doi.org/10.1038/s41746-023-00838-3

Jaber, M. M., Abd, S. K., Shakeel, P. M., Burhanuddin, M. A., Mohammed, M. A., & Yussof, S. (2020). A telemedicine tool framework for lung sounds classification using ensemble classifier algorithms. Measurement: Journal of the

International Measurement Confederation, 162. https://doi.org/10.1016/j.measurement.2020.107883

Jácome, C., Ravn, J., Holsbø, E., Aviles-Solis, J. C., Melbye, H., & Bongo, L. A. (2019). Convolutional neural network for breathing phase detection in lung sounds. Sensors (Switzerland), 19(8). https://doi.org/10.3390/s19081798

Jindal, V., Agarwal, V., & Kalaivani, S. (2018). Respiratory sound analysis for detection of pulmonary diseases. Proceedings of 2018 IEEE Applied Signal Processing Conference, ASPCON 2018, 293–296. https://doi.org/10.1109/ASPCON.2018.8748284

Kababulut, F. Y., Gürkan Kuntalp, D., Düzyel, O., Özcan, N., & Kuntalp, M. (2023). A New Shapley-Based Feature Selection Method in a Clinical Decision Support System for the Identification of Lung Diseases. Diagnostics, 13(23). https://doi.org/10.3390/diagnostics13233558

Kababulut, F. Y., Gürkan Kuntalp, D., & Kuntalp, M. (2021). Healthy-unhealthy classification using respiratory sounds and shapley values of features. Second International Artificial Intelligence in Health Congress, 263–274.

Kahya, Y., Yeginer, M., & Bilgic, B. (2006). Classifying Respiratory Sounds with Different Feature Sets. 2006 International Conference of the IEEE Engineering in Medicine and Biology Society, 2856–2859.

Karakış, G. P. (2020, May 13). What is Bronchiolitis? https://www.florence.com.tr/bronsiolit-nedir#:~:text=Bron%C5%9Fiolit%2C%20akci%C4%9Ferlerin%20k%C3%BC%C3%A7%C3%BCk%20havayollar%C4%B1nda%20(bron%C5%9Fiollerde,riskinin%20olu%C5%9Fu%20nedeniyle%20aileleri%20endi%C5%9Felendirir.

Li, T. F., & Chang, S.-C. (2007). Speech recognition of mandarin syllables using both linear predict coding cepstra and Mel frequency cepstra. 19th Conference on Computational Linguistics and Speech Processing, 379–390.

Liang, M., & Hu, X. (2015). Recurrent Convolutional Neural Network for Object Recognition. 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 3367–3375.

Liu, C., Springer, D., Li, Q., Moody, B., Juan, R. A., Chorro, F. J., Castells, F., Roig, J. M., Silva, I., Johnson, A. E. W., Syed, Z., Schmidt, S. E., Papadaniil, C. D.,

Hadjileontiadis, L., Naseri, H., Moukadem, A., Dieterlen, A., Brandt, C., Tang, H., … Clifford, G. D. (2016). An open access database for the evaluation of heart sound algorithms. Physiological Measurement, 37(12), 2181–2213. https://doi.org/10.1088/0967-3334/37/12/2181

Majidnezhad, V., & Kheidorov, I. (2013). A Novel Method for Feature Extraction in Vocal Fold Pathology Diagnosis. Wireless Mobile Communication and Healthcare. MobiHealth 2012, 61, 96–105. https://doi.org/10.1007/978-3-642-37893-5

Maltepe Hospital. (2023, July 16). What Should We Do to Protect From Pneumonia? https://maltepehastanesi.com.tr/tr/makaleler/detay/zaturreden-korunmak-icin-neler-yapmaliyiz#:~:text=Ama%2C%20hastal%C4%B1k%20%C3%A7o%C4%9Funlukla%2C%20hastan%C4%B1n%20kendi,zay%C4%B1f%20d%C3%BC%C5%9Fm%C3%BC%C5%9F%20ki%C5%9Filerde%20zat%C3%BCrre%20olu%C5%9Fturur.

Manir, S. B., Karim M., & Kiber, M. A. (2020). Assessment of Lung Diseases from Features Extraction of Breath Sounds Using Digital Signal Processing Methods. 2020 Emerging Technology in Computing, Communication and Electronics (ETCCE), 1–6.

Martínez-Alajarín, J., López-Candel, J., & Ruiz-Merino, R. (2007). Classification and Diagnosis of Heart Sounds and Murmurs Using Artificial Neural Networks. Bio-Inspired Modeling of Cognitive Tasks. IWINAC 2007. Lecture Notes in Computer Science, 303–312.

Mathworks. (2023). Cluster Data with a Self-Organizing Map. https://www.mathworks.com/help/deeplearning/gs/cluster-data-with-a-self-organizing-map.html

Memorial Hospitals. (2023, September 13). What is Asthma? What are the Symptoms of Asthma? https://www.memorial.com.tr/hastaliklar/astim#:~:text=Ast%C4%B1m%3B%20solunum%20yollar%C4%B1n%C4%B1n%20daralmas%C4%B1%2C%20%C5%9Fi%C5%9Fmesi,%C3%A7ektiren%20kronik%20bir%20akci%C4%9Fer%20hastal%C4%B1%C4%9F%C4%B1d%C4%B1r.

Messaoudi, I., Oukhai, O., Oueslati, A. E., & Lachiri, Z. (2022). Comparative Analysis of Linear Regression and Logistic Regression for the COVID-19 Detection from CT-Scans. 2022 IEEE Information Technologies and Smart Industrial Systems, ITSIS 2022. https://doi.org/10.1109/ITSIS56166.2022.10118393

Mor, R., Kushnir, I., Meyer, J.-J., Ekstein, J., & Ben-Dov, I. (2007). Breath Sound Distribution Images of Patients With Pneumonia and Pleural Effusion. Respiratory Care, 52(12), 1753–1760.

Moussavi, Z. K., Leopando, M. T., & Rempel, G. R. (1998). Automated detection of respiratory phases by acoustical means. Proceedings of the 20th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 21–24. https://doi.org/10.1109/iembs.1998.745812

Mqadi, N. M., Naicker, N., & Adeliyi, T. (2021). Solving Misclassification of the Credit Card Imbalance Problem Using near Miss. Mathematical Problems in Engineering, 2021. https://doi.org/10.1155/2021/7194728

Mukherjee, H., Sreerama, P., Dhar, A., Obaidullah, S. M., Roy, K., Mahmud, M., & Santosh, K. C. (2021). Automatic Lung Health Screening Using Respiratory Sounds. Journal of Medical Systems, 45(2). https://doi.org/10.1007/s10916-020-01681-9

Murphy, RL., Vyshedskiy, A., Power-Charnitsky, V.-A., Bana, D. S., Marinelli, P. M., Wong-Tse, A., & Paciej, R. (2004). Automated Lung Sound Analysis in Patients With Pneumonia. Respiratory Care, 49(12), 1490–1497.

Naqvi, S. Z. H., & Choudhry, M. A. (2020). An automated system for classification of chronic obstructive pulmonary disease and pneumonia patients using lung sound analysis. Sensors (Switzerland), 20(22), 1–23. https://doi.org/10.3390/s20226512

Nigam, V., & Priemer, R. (2005). Simplicity Based Gating of Heart Sounds. 48th Midwest Symposium on Circuits and Systems, 1298–1301.

Nilanon, T., Yao, J., Hao, J., Purushotham, S., & Liu, Y. (2016). Normal / Abnormal Heart Sound Recordings Classification Using Convolutional Neural Network. 2016 Computing in Cardiology Conference (CinC), 585–588.

Ntalampiras, S. (2020). Collaborative framework for automatic classification of respiratory sounds. IET Signal Processing, 14(4), 223–228. https://doi.org/10.1049/iet-spr.2019.0487

Oletic, D., & Bilas, V. (2016). Energy-efficient respiratory sounds sensing for personal mobile asthma monitoring. IEEE Sensors Journal, 16(23). https://doi.org/10.1109/JSEN.2016.2585039

Oliveira, J., Renna, F., Mantadelis, T., & Coimbra, M. (2019). Adaptive Sojourn Time HSMM for Heart Sound Segmentation. IEEE Journal of Biomedical and Health Informatics, 23(2), 642–649. https://doi.org/10.1109/JBHI.2018.2841197

Olmez, O., & Dokur, umray. (2003). Classification of heart sounds using an artificial neural network. Pattern Recognition Letters, 24(1), 617–629. www.elsevier.com/locate/patrec

Olvera-Montes, N., Reyes, B., Charleston-Villalobos, S., Gonzalez-Camarena, R., Mejiaavila, M., Dorantes-Mendez, G., Reulecke, S., & Aljama-Corrales, T. A. (2018). Detection of Respiratory Crackle Sounds via an Android Smartphone-based System. Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS, 2018-July. https://doi.org/10.1109/EMBC.2018.8512672

Ono, H., Taniguchi, Y., Shinoda, K., Sakamoto, T., Kudoh, S., & Gemma, A. (2009). Evaluation of the usefulness of spectral analysis of inspiratory lung sounds recorded with phonopneumography in patients with interstitial pneumonia. Journal of Nippon Medical School, 76, 67–75.

Palaniappan, R., & Sundaraj, K. (2013). Respiratory sound classification using cepstral features and support vector machine. 2013 IEEE Recent Advances in Intelligent Computational Systems (RAICS), 132–136.

Palaniappan, R., Sundaraj, K., & Sundaraj, S. (2014). A comparative study of the svm and k-nn machine learning algorithms for the diagnosis of respiratory pathologies using pulmonary acoustic signals. BMC Bioinformatics, 15(1). https://doi.org/10.1186/1471-2105-15-223

Papadaniil, C. D., & Hadjileontiadis, L. J. (2014). Efficient heart sound segmentation and extraction using ensemble empirical mode decomposition and

kurtosis features. IEEE Journal of Biomedical and Health Informatics, 18(4), 1138–1152. https://doi.org/10.1109/JBHI.2013.2294399

Perna, D., & Tagarelli, A. (2019, July 11). Deep auscultation: Predicting respiratory anomalies and diseases via recurrent neural networks. 2019 IEEE 32nd International Symposium on Computer-Based Medical Systems (CBMS), 50–55. http://doi.org/ 10.1109/CBMS.2019.00020

Pham, L., McLoughlin, I., Phan, H., Tran, M., Nguyen, T., & Palaniappan, R. (2020). Robust Deep Learning Framework For Predicting Respiratory Anomalies and Diseases. 2020 42nd Annual International Conference of the IEEE Engineering in Medicine \& Biology Society (EMBC), 164–167. http://doi.org/ 10.1109/EMBC44109.2020.9175704

Polat, S. K. (2022, July 28). Bronchiectasis Disease. https://www.memorial.com.tr/hastaliklar/bronsektazi-hastaligi

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), 621–624.

Priftis, K. N., Hadjileontiadis, L. J., & Everard, M. L. (2018). Breath Sounds From Basic Science to Clinical Practice. Springer.

Putrada, A. G., Alamsyah, N., Pane, S. F., & Fauzan, M. N. (2022). XGBoost for IDS on WSN Cyber Attacks with Imbalanced Data. ISESD 2022 - 2022 International Symposium on Electronics and Smart Devices, Proceeding. https://doi.org/10.1109/ISESD56103.2022.9980630

Rajabi, A., & Garibay, O. O. (2022). TabFairGAN: Fair Tabular Data Generation with Generative Adversarial Networks. Machine Learning and Knowledge Extraction, 4(2), 488–501. https://doi.org/10.3390/make4020022

Randhawa, S. K., & Singh, M. (2015). Classification of Heart Sound Signals Using Multi-modal Features. Procedia Computer Science, 58, 165–171. https://doi.org/10.1016/j.procs.2015.08.045

Rao, A., Huynh, E., Royston, T. J., Kornblith, A., & Roy, S. (2019). Acoustic Methods for Pulmonary Diagnosis. IEEE Reviews in Biomedical Engineering, 12, 221–239. https://doi.org/10.1109/RBME.2018.2874353

Riella, R. J., Nohama, P., & Maia, J. M. (2009). Method for automatic detection of wheezing in lung sounds. In Brazilian Journal of Medical and Biological Research (Vol. 42, Issue 7). www.bjournal.com.br

Rietveld, S., Oud, M., & Dooijes, E. H. (1999). Classification of Asthmatic Breath Sounds: Preliminary Results of the Classifying Capacity of Human Examiners versus Artificial Neural Networks. In Computers and Biomedical Research (Vol. 32). http://www.idealibrary.comon

Rocha, B. M., Filos, D., Mendes, L., Vogiatzis, I., Perantoni, E., Kaimakamis, E., Natsiavas, P., Oliveira, A., Jácome, C., Marques, A., Paiva, R. P., Chouvarda, I., Carvalho, P., & Maglaveras, N. (2017). A Respiratory Sound Database for the Development of Automated Classification. Precision Medicine Powered by PHealth and Connected Health. ICBHI 2017. https://bhichallenge.med.auth.gr/

Rocha, B. M., Pessoa, D., Marques, A., Carvalho, P., & Paiva, R. P. (2021). Automatic classification of adventitious respiratory sounds: A (un)solved problem? Sensors (Switzerland), 21(1), 1–19. https://doi.org/10.3390/s21010057

Rubin, J., Abreu, R., Ganguli, A., Nelaturi, S., Matei, I., & Sricharan, K. (2016). Classifying Heart Sound Recordings using Deep Convolutional Neural Networks and Mel-Frequency Cepstral Coefficients. 2016 Computing in Cardiology Conference (CinC), 813–816.

Saldanha, J., Chakraborty, S., Patil, S., Kotecha, K., Kumar, S., & Nayyar, A. (2022). Data augmentation using Variational Autoencoders for improvement of respiratory disease classification. PLoS ONE, 17(8 August). https://doi.org/10.1371/journal.pone.0266467

Samadianfard, S., Kargar, K., Shadkani, S., Hashemi, S., Abbaspour, A., & Safari, M. J. S. (2022). Hybrid models for suspended sediment prediction: optimized random forest and multi-layer perceptron through genetic algorithm and stochastic gradient descent methods. Neural Computing and Applications, 34(4), 3033–3051. https://doi.org/10.1007/s00521-021-06550-1

Schölkopf, B., Tsuda, K., & Vert, J. V. (2004). A primer on kernel methods. In Kernel Methods in Computational Biology. The MIT Press.

Sengupta, N., Sahidullah, M., & Saha, G. (2016). Lung sound classification using cepstral-based statistical features. Computers in Biology and Medicine, 75, 118–129. https://doi.org/10.1016/j.compbiomed.2016.05.013

Shapley, L. S. (1951). Notes on the N-Person Game — II: The Value of an N-Person Game. RAND Corporation.

Shuvo, S. B., Ali, S. N., Swapnil, S. I., Hasan, T., & Bhuiyan, M. I. H. (2021). A Lightweight CNN Model for Detecting Respiratory Diseases from Lung Auscultation Sounds Using EMD-CWT-Based Hybrid Scalogram. IEEE Journal of Biomedical and Health Informatics, 25(7), 2595–2603. https://doi.org/10.1109/JBHI.2020.3048006

Sim, T., Choi, S., Kim, Y., Youn, S. H., Jang, D. J., Lee, S., & Chun, C. J. (2022). Explainable AI (XAI)-based input variable selection methodology for forecasting energy consumption. Electronics (Switzerland), 11(18). https://doi.org/10.3390/electronics11182947

Singh, P. P., & Rani, P. (2014). An Approach to Extract Feature using MFCC. IOSR Journal of Engineering (IOSRJEN) Www.Iosrjen.Org ISSN, 04, 1–21. www.iosrjen.org

Srivastava, A., Jain, S., Miranda, R., Patil, S., Pandya, S., & Kotecha, K. (2021). Deep learning based respiratory sound analysis for detection of chronic obstructive pulmonary disease. PeerJ Computer Science, 7, 1–22. https://doi.org/10.7717/PEERJ-CS.369

Stoica, P., & Moses, R. L. (2005). Spectral Analysis of Signals (T. Robbins, Ed.). Pearson/Prentice Hall.

Tabatabaei, S. A. H., Fischer, P., Schneider, H., Koehler, U., Gross, V., & Sohrabi, K. (2021). Methods for Adventitious Respiratory Sound Analyzing Applications Based on Smartphones: A Survey. In IEEE Reviews in Biomedical Engineering (Vol. 14, pp. 98–115). Institute of Electrical and Electronics Engineers Inc. https://doi.org/10.1109/RBME.2020.3002970

Van Der Maaten, L., & Hinton, G. (2008). Visualizing Data using t-SNE. Article in Journal of Machine Learning Research, 6, 2579–2605. https://www.researchgate.net/publication/228339739

Waitman, L. R., Clarkson, K. P., Barwise, J. A., King, P. H., Lr, W., Kp, C., & Ja, B. (2000). Representation and classification of breath sounds recorded in an intensive care setting using neural networks. Journal of Clinical Monitoring and Computing, 16, 95–105.

Wang, W., Li, S., Yang, J., Liu, Z., & Zhou, W. (2016). Feature Extraction of Underwater Target in Auditory Sensation Area Based on MFCC. 2016 IEEE/OES China Ocean Acoustics (COA), 1–6.

Xu, L., Skoularidou, M., Cuesta-Infante, A., & Veeramachaneni, K. (2019). Modeling Tabular Data using Conditional GAN. Advances in Neural Information Processing Systems 32 (NeurIPS 2019). https://github.com/DAI-Lab/CTGAN

Zhang, K., Zhang, J., Xu, P. D., Gao, T., & Gao, D. W. (2022). Explainable AI in deep reinforcement learning models for power system emergency control. IEEE Transactions on Computational Social Systems, 9(2). https://doi.org/10.1109/TCSS.2021.3096824

Zhang, Y., Ayyar, S., Chen, L.-H., & Li, E. J. (2016). Segmental Convolutional Neural Networks for Detection of Cardiac Abnormality With Noisy Heart Sound Recordings. http://arxiv.org/abs/1612.01943

Zhao, Z., Kunar, A., Birke, R., & Chen, L. Y. (2021). CTAB-GAN: Effective Table Data Synthesizing. The 13th Asian Conference on Machine Learning, 97–112.

Zhong, L., Wan, J., Huang, Z., Cao, G., & Xiao, B. (2013). Heart Murmur Recognition Based on Hidden Markov Model. Journal of Signal and Information Processing, 04(02), 140–144. https://doi.org/10.4236/jsip.2013.42020