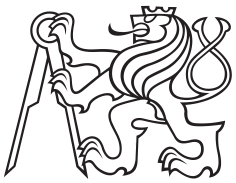


Czech
Technical
University
in Prague

F3

Faculty of Electrical Engineering
Department of Circuit Theory

Master Thesis

# Interpretable Lung Perfusion Imaging with Feature-Based Modelling of EIT's Cardiac-Related Signal

Annamária Miheličová

Supervisor: Diogo Silva, M.Sc. (RWTH Aachen University, Germany)
Second supervisor: Ing. Jan Havlík, Ph.D.
Field of study: Medical Electronics and Bioinformatics
Subfield: Medical Instrumentation
May 2024

# MASTER‘S THESIS ASSIGNMENT

## I. Personal and study details

Student's name: **Miheličová Annamária** Personal ID number: **483455**

Faculty / Institute: **Faculty of Electrical Engineering**

Department / Institute: **Department of Circuit Theory**

Study program: **Medical Electronics and Bioinformatics**

Specialisation: **Medical Instrumentation**

## II. Master’s thesis details

Master’s thesis title in English:

**Interpretable Lung Perfusion Imaging with Feature-Based Modelling of EIT’s Cardiac-Related Signal**

Master’s thesis title in Czech:

**Interpretovatelné zobrazování perfuze plic s modelováním na základě charakteristických rysů souvisejících se srdečním signálem EIT**

Guidelines:

- Implementation of extraction routines for a large number of varied relevant features
- Feature extraction and assembly of a large feature dataset
- Statistical analysis for the selection and interpretation of important features, and identification of feature interactions
- Appropriate application and training of candidate data-driven models
- Interpretation of test results and iterative refactoring of the processing pipeline

Bibliography / sources:

[1] S. Leonhardt and B. Lachmann, “Electrical impedance tomography: the holy grail of ventilation and perfusion monitoring?,” Intensive Care Med., vol. 38, no. 12, pp. 1917–1929, Dec. 2012, doi: 10.1007/s00134-012-2684-z.
[2] C. Putensen, B. Hentze, S. Muenster, and T. Muders, “Electrical Impedance Tomography for Cardio-Pulmonary Monitoring,” J. Clin. Med., vol. 8, no. 8, p. 1176, Aug. 2019, doi: 10.3390/jcm8081176
[3] J. Solà, A. Adler, A. Santos, G. Tusman, F. S. ^ Sipmann, and S. H. Bohm, “Non-invasive monitoring of central blood pressure by electrical impedance tomography: first experimental evidence,” Med. Biol. Eng. Comput., vol. 49, no. 4, pp. 409–415, Apr. 2011, doi: 10.1007/s11517-011-0753-z
[4] C. A. Grant, T. Pham, J. Hough, T. Riedel, C. Stocker, and A. Schibler, “Measurement of ventilation and cardiac related impedance changes with electrical impedance tomography,” Crit. Care, vol. 15, no. 1, p. R37, 2011, doi: 10.1186/cc9985.
[5] I. Frerichs et al., “Assessment of Changes in Distribution of Lung Perfusion by Electrical Impedance Tomography,” Respiration, vol. 77, no. 3, pp. 282–291, 2009, doi: 10.1159/000193994.
[6] M. Kircher et al., “Regional Lung Perfusion Analysis in Experimental ARDS by Electrical Impedance and Computed Tomography,” IEEE Trans. Med. Imaging, vol. 40, no. 1, pp. 251–261, Jan. 2021, doi: 10.1109/TMI.2020.3025080

Name and workplace of master’s thesis supervisor:

**Diogo Silva, MSc. RWTH Aachen University**

Name and workplace of second master’s thesis supervisor or consultant:

**Ing. Jan Havlík, Ph.D. Department of Circuit Theory, FEE**

Date of master’s thesis assignment: **31.08.2023** Deadline for master's thesis submission: **24.05.2024**

Assignment valid until: **16.02.2025**

| Diogo Silva, MSc. | doc. Ing. Radoslav Bortel, Ph.D. | prof. Mgr. Petr Páta, Ph.D. |
|---|---|---|
| Supervisor’s signature | Head of department’s signature | Dean’s signature |

## III. Assignment receipt

The student acknowledges that the master's thesis is an individual work. The student must produce her thesis without the assistance of others, with the exception of provided consultations. Within the master's thesis, the author must state the names of consultants and include a list of references.

| . | |
|---|---|
| Date of assignment receipt | Student's signature |

# Acknowledgements

I would like to express my gratitude to all those who supported me during my double-degree studies at CTU in Prague and RWTH Aachen. Specifically, I am thankful to my supervisor, M.Sc. Diogo Silva, for his guidance, support, and feedback throughout the entire process of writing this thesis. I extend my appreciation to my CTU supervisor, Ing. Jan Havlík, Ph.D., for his continuous support not only during the thesis but also throughout my entire master's studies. Most importantly, I want to express deep gratitude to my family and boyfriend for their unwavering love and support in my life.

Some computational resources were provided by the e-INFRA CZ project (ID:90254), supported by the Ministry of Education, Youth and Sports of the Czech Republic. Thank you.


# Declaration

I declare that the presented work was developed independently and that I have listed all sources of information used within it in accordance with the methodical instructions for observing the ethical principles in the preparation of university theses.

In Prague, on 24.5.2024

.....
Annamária Miheličová

# Abstract

Obtaining the lung perfusion distribution in a continuous, non-invasive way is considered a promising method, as the current gold standard, traditional radiation-based imaging method, represents an invasive and potentially harmful solution. The aim of this thesis is to develop a data-driven statistical model capable of assessing perfusion in real-time and non-invasively by focusing on leveraging the cardiac component of the electrical impedance tomography signal and utilizing extracted features from the signal.

The thesis employs two main statistical tasks: regression to quantify lung perfusion and classification to assess lung blockages. Models are trained on real animal data with varying perfusion insufficiencies, with the flow derived from the enhanced signal after saline contrast injection, as a reference. Cross-validation techniques are used for unbiased evaluation of results, revealing variability in data among animals that affects model performance. Interpretable models, ranging from simple logistic regression to variants of the generalized additive model are implemented to provide a solution and are compared with black-box models like Random Forests and XGBoost.

Although black-box models generally slightly outperform interpretable ones, interpretable models show promising results in both regression and classification tasks compared to a single-feature linear model comparable to the state-of-the-art, termed as a baseline model.

This thesis successfully investigates the cardiac-related signal in the context of lung perfusion evaluation, highlighting its capabilities and limitations. The results and future investigations may lead to the development of a non-invasive and real-time approach to assess lung perfusion.



**Keywords:** electrical impedance tomography, cardiac-related signal, interpretable models, lung perfusion



# Abstrakt

Získanie distribúcie perfúzie pľúc kontinuálnym, neinvazívnym spôsobom sa považuje za sľubnú metódu, keďže súčasný zlatý štandard, tradičná zobrazovacia metóda založená na žiarení, predstavuje invazívne a potenciálne škodlivé riešenie. Cieľom tejto práce je vyvinúť štatistický model založený na údajoch, ktorý dokáže posúdiť perfúziu neinvazívne a v reálnom čase, pričom sa zameriava na využitie srdcovej zložky signálu z elektrickej impedančnej tomografie a využitie extrahovaných znakov zo signálu.

V práci sa využívajú dve hlavné štatistické úlohy: regresia na kvantifikáciu perfúzie pľúc a klasifikácia na posúdenie ich blokády. Modely sa trénujú na skutočných údajoch zvierat s rôznou mierou perfúzie, pričom ako referencia slúži prietok odvodený zo signálu po podaní kontrastného roztoku. Pre objektívne vyhodnotenie výsledkov sa používajú techniky krížovej validácie, ktoré odhaľujú variabilitu dat medzi zvieratami, ktorá ovplyvňuje kvalitu modelov. Interpretovateľné modely, od jednoduchej logistickej regresie až po zovšeobecnené aditívne modely sú implementované s cieľom poskytnúť riešenie a porovnávajú sa s black-box modelmi, ako napríklad Random Forests a XGBoost.

Hoci black-box modely všeobecne mierne prekonávajú interpretovateľné modely, aj oni vykazujú sľubné výsledky v regresii a klasifikácii v porovnaní s jednoprízнakovým lineárnym modelom porovnateľným so state-of-the-art modelom, označeným ako baseline model.

Táto práca úspešne skúma signál súvisiaci so srdcom v kontexte perfúzie pľúc, pričom poukazuje na jeho možnosti a limity. Výsledky a budúce výskumy môžu viesť k vývoju neinvazívneho prístupu na určenie perfúzie pľúc v reálnom čase.



**Kľúčové slová:** elektrická impedančná tomografia, signál súvisiaci so srdcom, interpretovateľné modely, pľúcna perfúzia

# Contents

# Figures

# Tables

# Chapter 1

# Introduction

The human respiratory system is essential for exchanging oxygen and carbon dioxide, and any disruption can have serious, life-threatening consequences. Measuring lung perfusion can be a useful way to identify the right treatment.

Electrical impedance tomography (EIT) is a radiation-free imaging technique that allows for the assessment of electrical impedance within the thoracic cavity, providing insight into lung dynamics, blood flow distribution, and the transmission of cardiac-related signals through lung tissue [1]. This modality has several advantages, such as being non-invasive and having no fatal effect on the patient being examined. Additionally, it is cost-effective and provides real-time results, meaning that any changes in function can be detected immediately [2].

In order to acquire EIT images, electrodes are placed evenly around the chest. A small current is then injected into the body and the voltage that is produced is measured [1]. The alterations in impedance caused by breathing can be detected in the area of the heart and lungs.

There are two techniques for measuring lung perfusion in both areas. The invasive approach involves injecting a contrast agent into the bloodstream. The difference in conductivity between the agent and the blood allows tracking of changes in impedance, which are related to perfusion [3]. Unfortunately, this method is not suitable for continuous monitoring due to its invasive nature.

In comparison to the invasive approach, the non-invasive is based on the electrical activity of the heart, which leads to alterations in the impedance of the heart and lungs. This is because blood has a higher conductivity than tissue, which results in a lower impedance measurement. Changes are caused by blood flow through the organs [1]. Although the effect of the cardiac signal on lung perfusion is still being studied, a recent study [4] provided clear evidence of a link between the two.

This thesis investigates the connection between the above mentioned organs by concentrating on discovering suitable features that can be used to interpret the lung perfusion from the cardiac signal based on EIT measurement. Firstly, a dataset of extracted features from the recorded data is created. The dataset is then visualized to identify the features that are relevant for further processing. Subsequently, a statistical methods are chosen and implemented for the selection of significant features and evaluation of the results. Finally, the results of the statistical test are validated.

## 1.1 Structure of the thesis

This thesis is structured into five chapters. Chapter 2 explains the theoretical principles necessary for a comprehensive understanding of the topic. It covers the EIT method, IBS as the current gold standard method used as a reference, and the origin of CRS. Additionally, the statistical methods and models used are explained in the conclusion of this chapter. Chapter 3 focuses on explaining the methods and implementation of extraction routines for a large number of relevant features and an appropriate application and training of candidate data-driven models. The results are presented and evaluated in chapter 4, where the interpretation of test results and iterative refactoring of the processing pipeline is also discussed. Chapter 5 concludes the thesis, providing an overview of the entire process and providing insights for future directions.

# Chapter 2
# Theoretical Background

This chapter gives a brief overview of the fundamental theoretical concepts employed in the diploma thesis. Initially, electrical impedance tomography (EIT) is explored by explaining its measurement principle and related components. Subsequently, image reconstruction techniques utilizing both forward and inverse methods are described.

Following this, the chapter explains the monitoring of lung perfusion and the use of the cardiac component of the recordings (CRS). It also examines the indicator bolus signal (IBS) method, analyzing its advantages and disadvantages together with comparison to the EIT. Additionally, it describes the process of extracting ground-truth signals for comparative analysis purposes.

In the final section of the chapter, the fundamental principles of statistical analysis are presented. Initially, the distinction between regression and classification tasks are described. Then, the principles underlying the implemented statistical models for each task are explained. Lastly, the chapter discusses various evaluation metrics.

## 2.1 Electrical Impedance Tomography

Electrical impedance tomography (EIT) is a radiation-free two-dimensional imaging method utilized to determine electrical conductivity within the heart and thoracic region [5]. Its hardware components are relatively simple, making it a cost-effective alternative to other imaging modalities. Moreover, its non-invasive nature permits prolonged bedside monitoring. EIT generates images through the placement of 16 surface electrodes around the thorax, using the varying conductivity of blood and tissues during systole and diastole. This fluctuation in conductivity, attributed to the heart pumping action and the transition from diastole to systole, produces a measurable voltage across the electrodes [6, 5].

Despite its high temporal resolution (13-50 frames/s), this method suffers from poor spatial resolution [7]. In EIT, a small electrical current of 5 mA is applied at a frequency of 100 kHz to a pair of electrodes, while the voltage is measured on the remaining electrodes, as illustrated in Figure 2.1. The system then calculates both the amplitude and phase simultaneously [6]. Further details of EIT's measurement principle will be provided in the subsequent sections.

### 2.1.1 Bioimpedance

Bioimpedance refers to the electrical impedance, denoted by Z, present in biological tissues. Typically, it is assessed using the four-electrode method, where a small alternating current is applied to a pair of electrodes, while the voltage is measured on the remaining two. The impedance magnitude |Z| and the phase angle $\Phi$ are determined by the current frequency, as outlined in Equations 2.2 and 2.3 [8].

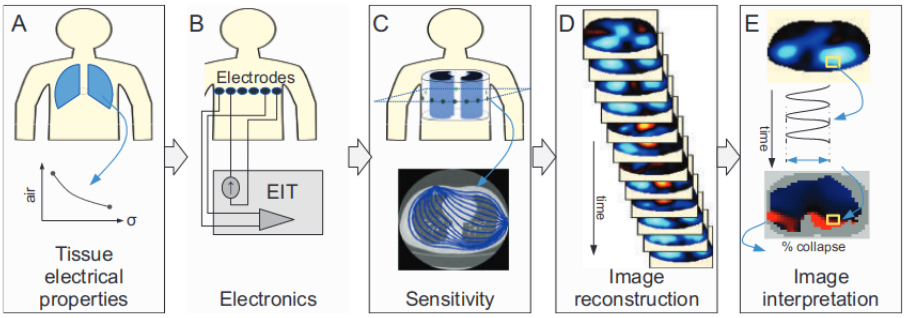



**Figure 2.1:** Overview of the steps in EIT image generation and interpretation [6].

Mathematically the electrical impedance is expressed by following equations

$$Z = R + jX_C, \tag{2.1}$$

$$|Z| = \sqrt{R^2 + X_C^2}, \tag{2.2}$$

$$\Phi = tan^{-1}\left(\frac{X_C}{R}\right). \tag{2.3}$$

Equation 2.1 shows that this quantity is complex, comprising both resistance (R) and reactance ($X_C$). It offers valuable insights into biological systems, particularly in the context of EIT, where it helps measure intrathoracic bioimpedance [1].

### 2.1.2 Adjacent Driven Configuration

The adjacent driven configuration is the most commonly used method for EIT measurement. It involves placing 16 electrodes in a belt configuration around the patient's chest, typically utilizing biocompatible materials that are washable for reuse. Typically, saline solutions are applied to improve contact between the electrode and the skin, with monitoring usually conducted in the fourth and sixth intercostal spaces for lung evaluation.

In this principle, a small alternating current (AC) is injected into one electrode pair (electrode 1 and electrode 2) at time $t$, as depicted in Figure 2.2, while the voltages are measured in all other electrodes. This results in $N - 3 = 13$ combinations [6, 7]. After the initial measurement, the injection current is transferred to another pair of electrodes (electrodes 2 and 3) at time $t + \Delta t$, and the voltages are measured again. To construct a complete EIT image, this process of rotating the injection and measurement electrodes around the thorax is repeated.

The AC current employed in this method is non-invasive, posing no harm to the human body as it is applied to the surface. Nonetheless, caution is advised when conducting measurements on patients with implanted electrical devices, such as cardiac pacemakers or defibrillators [1].

### 2.1.3 Image Reconstruction

Electrical Impedance Tomography (EIT) images are generated using image reconstruction techniques on the recorded frames. This approach relies on determining the distribution of electrical voltages within the thoracic cavity. Although the principle of rendering EIT images shares similarities with two-dimensional Computed Tomography (CT) scans, there are distinctions in the projection region. CT images are presented in slices, whereas EIT images are represented in regions, with the sensitivity region influenced by factors such as

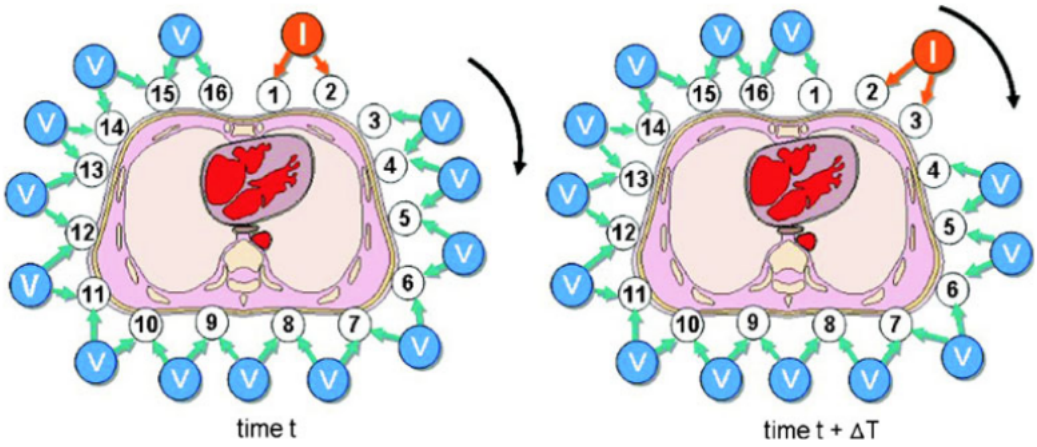



**Figure 2.2:** Adjacent driven configuration used in EIT measurement [7].

the size and shape of the thorax, the injected current, and the measured voltages [9]. This chapter explains the image reconstruction methods utilized in contemporary EIT practices. In contrast, CT imaging uses X-rays to produce a cross-sectional body image, a method considered more invasive than EIT.

### Forward Problem

The forward problem involves the propagation of an injected current through the body and the distribution of measured voltages given the electrical properties. The finite element method (FEM) stands out as the most widely used approach to solve this problem. However, this method requires the Laplace equation and the boundary conditions, which are often unknown, to obtain an accurate approximation for FEM [6, 10]. This is expressed by Equation 2.4.

$$\mathbf{K} \cdot \mathbf{\Phi} = \mathbf{J}, \tag{2.4}$$

where $\mathbf{K}$ represents the conductivity matrix of global coordinates of the mesh at a particular distribution, $\Phi$ denotes the potential matrix corresponding to each applied current pattern, and $\mathbf{J}$ signifies linearly independent patterns.

The solution of Equation 2.4 is known as an observation model, defined by Equation 2.5 as

$$\Phi = f(\rho), \tag{2.5}$$

where $\Phi$ denotes the calculated voltages with the observation model, and $\rho$ defines the state in which the observation occurs (which refers to the impedance distribution). This model neglects discretization errors, as the mesh is considered to be highly precise, as reported in [10].

### Inverse Problem

The inverse problem involves determining the internal impedance distribution that best fits the measured voltages, essentially reconstructing the EIT measurement. Equation 2.6 defines this as

$$\rho = f^{-1}(\Phi), \tag{2.6}$$

which is commonly referred to as an ill-posed problem. Image reconstruction can only provide an estimation of the internal distribution $\rho$.

The task of image reconstruction has significant difficulties, especially when interpreting lung EIT data [6]. Consequently, standardization efforts have led to the development of the GREIT (Graz consensus Reconstruction algorithm for EIT) algorithm, which currently represents the state-of-the-art method for EIT image reconstruction. GREIT is a mathematically defined and standardized technique for 2D EIT lung reconstruction, utilized in both clinical and research settings. It is expressed by the linear equation 2.7 as

$$\hat{\mathbf{x}} = \mathbf{R}\mathbf{y}, \tag{2.7}$$

where $\mathbf{y}$ represents a vector of EIT voltage differences, $\hat{\mathbf{x}}$ denotes a reconstructed voxel image, and $\mathbf{R}$ stands for the EIT reconstruction algorithm matrix [11].

### Time-difference Imagining

Time-difference imaging serves as a valuable tool to show conductivity changes in reconstructed EIT images. This technique compares impedance alterations with a baseline, allowing monitoring of dynamic processes such as lung perfusion, heartbeat, or real-time ventilation. Although the color-coding scheme for comparing organ stages lacks standardization, the rainbow-color scheme, depicted on the left side of Figure 2.3, is widely utilized. On the other hand, the black-blue-white scheme, shown on the right side of the same figure, is used to depict impedance changes, particularly encoding lung ventilation or perfusion. In addition, there are various other reconstruction modes, including absolute EIT, functional EIT, and EIT waveform analysis [12].

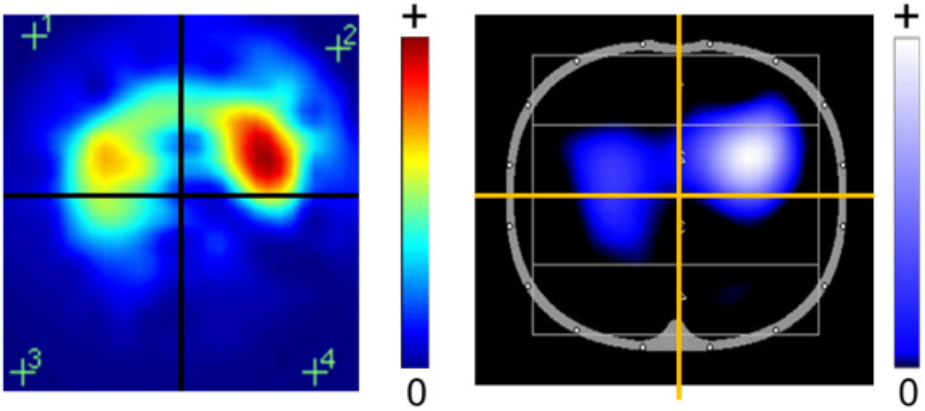



**Figure 2.3:** Color-coding used for representation of EIT images. Mostly, the red color represents the highest relative impedance during inspiration and the blue represents the lowest one during expiration. In the second case, black means that there is no impedance change, and white represents a strong impedance change [1].

## 2.2 Perfusion Monitoring

Gas exchange is one of the essential functions of the lungs, occurring primarily in structures called alveoli. Effective gas exchange is based on the ventilation and perfusion of these alveoli, making it a crucial aspect of lung function [13]. Consequently, measuring lung perfusion can help in diagnosing of various lung diseases.

However, finding a suitable method to address this challenge remains a subject of ongoing research. Functional imaging techniques such as CT, MRI, SPECT, and PET offer some

insights into ventilation distribution or perfusion, but do not allow frequent and continuous monitoring of lung perfusion. Furthermore, this problem predominantly affects patients who are often in bedside care [5]. Therefore, the development of appropriate portable and real-time imaging techniques and subsequent data analysis could be beneficial in their treatment.

To date, two potential solutions for measuring lung perfusion that meet the aforementioned requirements have been identified. The first involves an invasive approach known as the indicator bolus signal method (IBS), while the second utilizes a non-invasive monitoring method based on the measurement of cardiac activity (CRS) [1]. The working principles, its applications and limitations are described in this chapter.

### 2.2.1 Indicator Bolus Signal

The most common thermodilution technique used to monitor lung perfusion along with electrical impedance tomography (EIT) measurements is the indicator bolus signal (IBS). This method involves injecting a hypertonic saline solution, often referred to as a bolus (e.g., 10 ml of 5% NaCl) [6], into a central vein via a central venous catheter (CVC) and the bolus travels from the right heart to the lungs and subsequently returns to the left heart [14]. The dilution follows the gamma-variate course explained by Thompson et al. [15].

Traceability is possible by impedance variations resulting from the differing conductivities of the bolus and the blood. Brown et al. [16] proved that conductivity signals can be used for blood flow measurements. Based on this, in [14] the authors developed a method to directly calculate regional lung perfusion from the obtained EIT images.

A disadvantage of this method is its invasive nature, which requires the placement of a CVC for injection of bolus solution or regulation of blood pressure [6]. Moreover, it does not allow continuous monitoring. However, in the future, if an appropriate amount of contrast agent is found to be harmless for human use, the application of the IBS method may become feasible [1]. In the thesis, the method [14] is used as a state-of-the-art approach (IBS flow method) and is utilized as a reference for the evaluation of obtained results.

### 2.2.2 Cardiac Related Signal

The measurement of cardiac signal (CRS) can help with non-invasive monitoring of lung perfusion as during EIT measurements, it is possible to monitor cardiac activity not only within the heart but also in the lungs. In the heart region, the impedance increases during systole and decreases during diastole, affecting the lungs with a corresponding but delayed effect. In contrast, in the lungs, impedance decreases during systole and increases during diastole [17]. When impedance decreases, the conductivity increases, this process is shown in Figure 2.4.

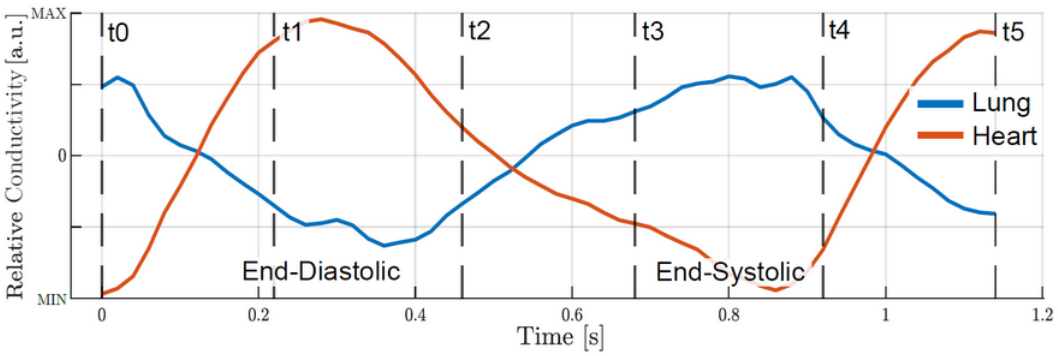



**Figure 2.4:** Conductivity change in lungs and heart during systole and diastole.

Apart from the origin in lung perfusion, the changes of regional impedance can be attributed to other factors. In [18] authors divided them into three categories: mechanical activity of the heart, such as changes in blood volume of the heart chambers, propagation of pressure pulses along blood vessels, for instance displacement of large arteries, and other sources such as cardiosynchronous movement of body.

There are two possible ways to obtain the CRS signal, however, both are challenging. The first extracts the CRS signal from the mixed EIT signal, where the second component represents the ventilation-related signal (VRS) that is approximately one order of magnitude higher than the CRS [7]. Moreover, the spectra of the CRS and VRS signals overlap and the application of traditional frequency filtration methods is not effective [1]. The other method obtains the CRS signal from the measurement during short apnea. The EIT signal is measured when the breathing is stopped, so it contains only the CRS signal. It is considered to be a more promising method, since the signal is clearer [1], however, according to [6] there are still some small ventilation-related artifacts and heart-lung coupling effects are absent. The signal used in this thesis was obtained by this method.

The main benefit of the CRS signal over IBS flow for monitoring of lung perfusion is its non-invasivity and the possibility of harmless continuous monitoring. The drawback of this method is in long-term monitoring, as there is a need for CRS separation. Apnea is possible only for a short time and reliable real-time filtration of the CRS and VRS components that would allow long-term monitoring is in ongoing research. Still this approach to obtaining the CRS signal during short-term apnea is adopted in this thesis. By extracting the signal features of different nature and implementation of statistical models, the regional lung perfusion can be investigated in real-time and noninvasively.

## 2.3 Statistical Methods

Statistical methods comprise the entire process, from formulating a hypothesis to either accepting or rejecting it. To establish this connection, it is essential to identify the data features that are relevant to explain the reference value. For finding this connection, it is necessary to find data features that are relevant to explain the reference quantity. Correlation tests are employed for this purpose. Once the relevant features are selected, an appropriate statistical model must be chosen to link them to the reference value. This is a process of supervised learning in which a reference variable is utilized in the model learning process. To evaluate the performance of the model, correct evaluation metrics are necessary (described in Chapter 2.5.1). Subsequent chapters will introduce the entire pipeline, from correlation tests to evaluation metrics.

### 2.3.1 Correlation Tests

Correlation tests are fundamental tools in statistics that are used to measure the strength and direction of the relationship between two variables. In the thesis, these two correlation tests are used: the Pearson correlation coefficient and the Spearman rank correlation coefficient.

#### Pearson's Correlation Test

Pearson's correlation is a statistical measure based on covariance that quantifies the strength and direction of a linear relationship between two continuous variables. Pearson's coefficient ($r$) obtains values in the range from -1 to 1 and is defined by Equation 2.8 as

$$r = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \bar{X})^2 \cdot \sum_{i=1}^{n}(Y_i - \bar{Y})^2}}, \tag{2.8}$$

where $r$ is the Pearson's coefficient, $X_i$ and $Y_i$ are individual data points, $\overline{X}$ and $\overline{Y}$ are the means of variables $X$ and $Y$. Based on the resulting correlation coefficient, the correlation strength is classified according to table 2.1 [19, 20].

| Pearson's correlation coefficient (r) value | Strength | Direction |
|---|---|---|
| Greater than 0.8 | Strong | Positive |
| Between 0.6 and 0.79 | Moderately strong | Positive |
| Between 0.3 and 0.59 | Fair | Positive |
| Less than 0.3 | Poor | Positive |
| 0 | No correlation | None |
| Less than –0.3 | Poor | Negative |
| Between –0.3 and -0.59 | Fair | Negative |
| Between -0.6 and -0.79 | Moderately strong | Negative |
| Grater than -0.8 | Strong | Negative |

**Table 2.1:** General correlation strength table for Pearson's correlation test. The table may vary between disciplines.

#### Spearman's Correlation Test

Spearman's correlation test is a nonparametric test and, in comparison with Pearson's correlation test, is more suitable for data that have violated normal distribution character. The Spearman's coefficient ($\rho$), defined by Equation 2.9 as

$$\rho = 1 - \frac{6\sum_{i=1}^{n} d_i^2}{n(n^2-1)}, \tag{2.9}$$

expresses the strength of the correlation and direction of the monotonic relationship between two variables and is therefore more suitable for variables that do not have a clear linear pattern [19, 21]. In Equation 2.9 $\rho$ is the Spearman's correlation coefficient, $d_i$ represents the difference between the ranks of the corresponding pairs of observations and $n$ is the number of pairs of observations.

### 2.3.2 Supervised Learning

Supervised learning is a form of learning in which the algorithm is trained to make predictions using labeled data. At the beginning, the algorithm receives a dataset of input features

along with the corresponding reference outputs. The learning process involves comparing the predicted results with the actual ones and adjusting based on the errors between the referenced and predicted values. The primary objective is to develop a model capable of predicting the output values based on a given set of features [22]. The complete supervised learning workflow is depicted in Figure 2.5.

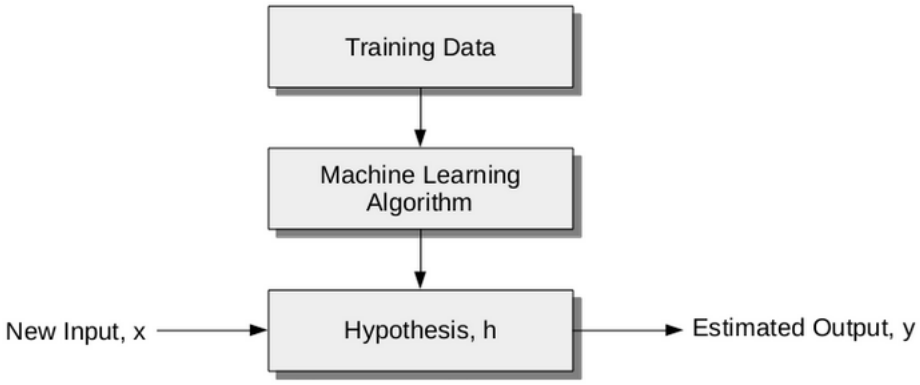



**Figure 2.5:** Supervised Learning Process [23].

This thesis addresses two tasks of supervised learning: classification and regression. Both approaches involve the construction of prediction models from the data [24]. The main difference between these two methods lies in their handling of values: regression deals with continuous or discrete values and predicts continuous numerical values, whereas classification aims to categorize each data point into specific categories or classes, resulting in categorical results [24, 25]. As shown in Figure 2.6, classification attempts to find a separation line that divides different types of values, while regression tries to find a line that minimizes the error of each point based on the squared difference between the predicted and measured values.

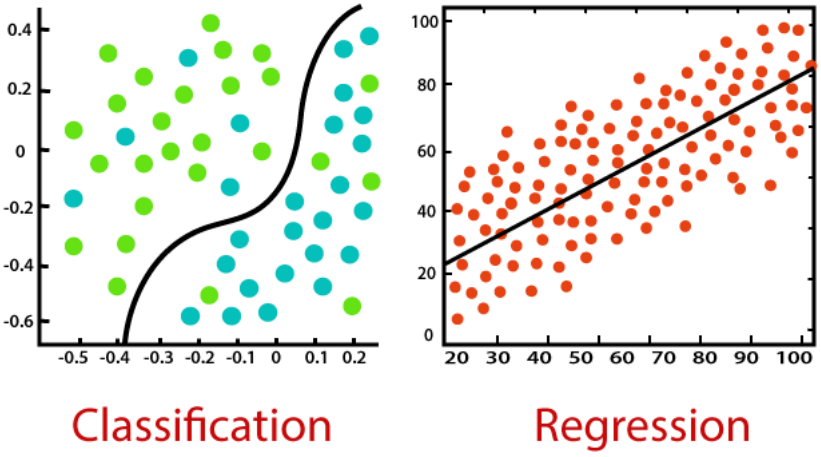



**Figure 2.6:** Illustration of regression and classification difference [25].

Regression analysis is a statistical method that explains the relationship between a dependent variable Y (target) and one or more independent variables $X$ (predictors), as defined by Equation 2.10 as

$$y_i = f(x_i, \beta) + \epsilon, \tag{2.10}$$

where $\beta$ represents the coefficients for each explanatory variable, denotes the error distribution

with a zero mean, also known as residuals. $y_i$ signifies dependant variable and $x_i$ represents explanatory variable [26]. The coefficients $\beta$ are unknown. To obtain the most accurate coefficient results, the least squares criterion is commonly employed. This approach estimates $\widehat{\beta}$ to minimize Residual Sum of Square (RSS) defined by Equation 2.11 as

$$RSS = e_1^2 + e_2^2 + ... + e_m^2 = \sum_{i=1}^{m}(y_i - f(x_i, \widehat{\beta}))^2. \tag{2.11}$$

Regression analysis allows an understanding of the correlations between variables and helps in predicting the results based on one or more independent variables. When employing regression, highly correlated independent variables (also referred to as features) should be removed from the dataset to mitigate the potential multicollinearity. Additionally, in regression, problems such as overfitting or underfitting may occur. Overfitting means that implemented algorithm performs well on training data, but has difficulties with testing data. However, underfitting occurs when the model is too simplistic and does not learn from the training data, resulting in poor model performance [27].

In classification, prediction involves accurately assigning the correct class to unknown data points, relying on the patterns learned from the training dataset. The objective is to develop a model that effectively accomplishes this task [28].

## 2.4 Regression and Classification Models

Statistical models enable data analysis, help to understand the relationship among the data, and provide insight into making conclusions from measurements. They give the possibility to model real problems and find a possible solution and can be categorized into interpretable and non-interpretable types. However, there is no exact mathematical definition of interpretability in machine learning. The concept is often described in terms of how well humans can understand the reasons behind a decision. For instance, according to [29], interpretability is the degree in which humans can understand the decision-making process. In this thesis, prioritizing highly interpretable methods is crucial due to the need to understand the feature selection process, especially in applications related to human health [30]. In addition, interpretable models are often taken to the real world and must be safe and tested [31]. However, non-interpretable models are also included for comparison purposes, as they often perform better than interpretable models, as demonstrated in Figure 2.7.

### 2.4.1 Generalized Additive Model

Generalized Additive Models or GAMs is an extension of the linear model introduced by [33]. This model allows generalizing of the Generalized Linear Model (GLM) including additivity of non-linear functions of the variables. It allows complex operations beyond linearity. The model is generally defined by equation 2.12 as

$$g(E(y)) = \beta + f_1(x_1) + f_2(x_2) + ... + f_m(x_m), \tag{2.12}$$

where $g(E(Y))$ is the link function that links the expected value to the predictor variables $x_1, x_2, \ldots, x_m$, $\beta$ is an intercept and $f_1(x_1), f_2(x_2), ..., f_m(x_m)$ are smooth functions of the predictor variables $x_1, x_2, \ldots, x_m$. The advantage of the model is its possibility of including multiple non-linear or smooth functions of selected level of smoothness, also called splines [34]. This approach is known as backfitting. The method fits a model using multiple predictors, iteratively updating the fit for each predictor while keeping the others fixed [28].

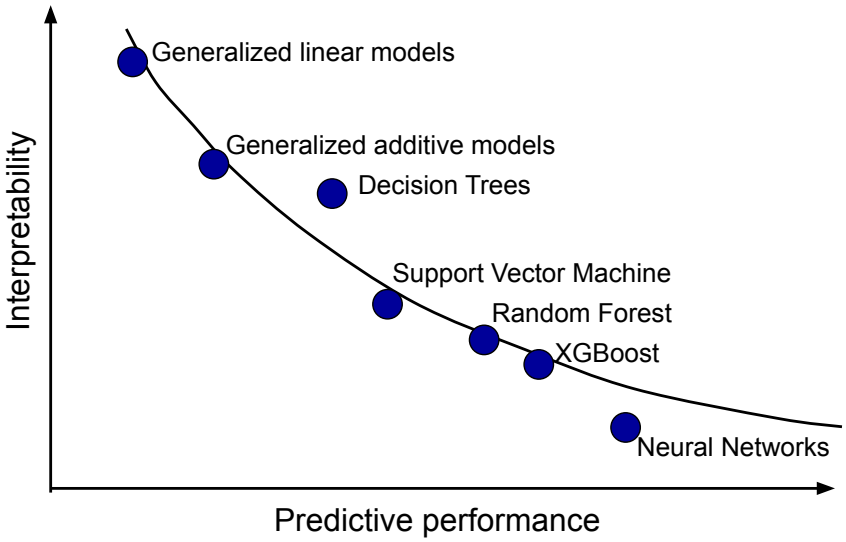



**Figure 2.7:** Interpretability vs. performance trade-off by statistical models, adopted from [32].

Moreover, the GAMs works with various distributions for the response variable that can be set. The method involves link functions to measure the impact of predictor variables on dependent regressors [28]. There are several distributions, such as the normal, gamma, or Poisson distribution. The term distribution expresses the type of error distribution. Among available link functions belong [35, 34]:

- Log link: $f(z) = log(z)$
- Inverse link: $f(z) = \frac{1}{z}$
- Identity link: $f(z) = z$
- Logit link: $f(z) = log(\frac{z}{1-z})$

### 2.4.2 Logistic Regression

Logistic regression is an appropriate solution for classification and is considered as an interpretable model. Although the structure of this model is simple, it performs well, especially when the relation between predicted and target variables is relatively linear. The logistic regression model employs a logistic function, also referred to as the sigmoid function, which is shown in Figure 2.8. This function squeezes the output of the linear equation within the range of 0 to 1 as demonstrated by [31] and expressed by Equation 2.13 as

$$\log\left(\frac{p(X)}{1-p(X)}\right) = \beta_0 + \beta_1 X. \tag{2.13}$$

The left side of the equation 2.13 is called a logistic function and has a linear relationship with X. In contrast to linear regression, the change in $X$ does not correspond to the change in $p(x)$, but the logarithmic odds change by a multiple of $\beta_1$. $\beta_0$ represents the intercept [28].

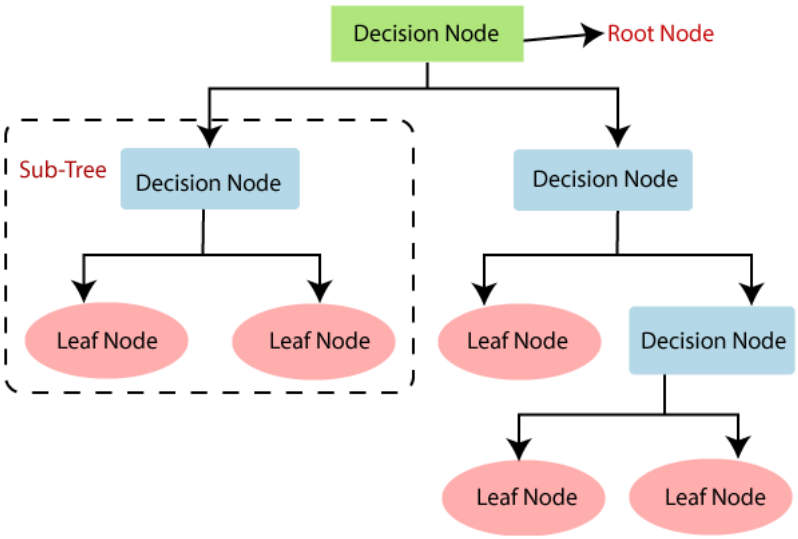



**Figure 2.9:** Graphical representation of Decision Tree [31].

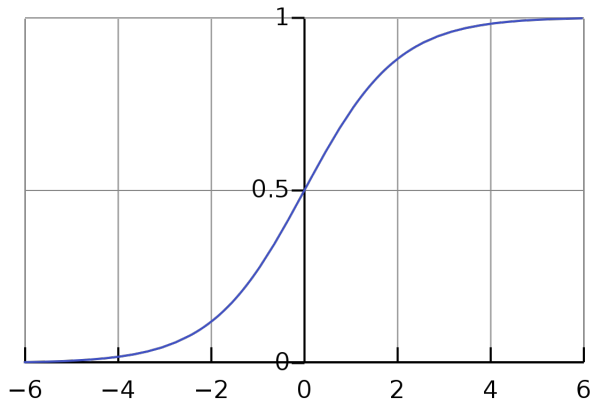



**Figure 2.8:** The logistic function. It outputs numbers between 0 and 1. At input 0, it outputs 0.5 [31].

Logistic models provide several benefits. They offer the probability of a data point belonging to a particular class based on the observed data. Logistic models are also resilient to outliers, restricting the output to values between 0 and 1. Furthermore, logistic regression is effective for handling imbalanced data, making it suitable for classification tasks [31].

### 2.4.3 Decision Tree Method

Decision tree models, like the methods mentioned above, are highly interpretable. Their key advantage is in addressing the nonlinear relationship between features and output, as well as interactions among features. These models have a hierarchical structure, as shown in the Firgure 2.9 where the starting point is called the root node, which then branches out into subsets (subtrees). The terminal subset is called the leaf node [31]. The root is whole dataset, branches introduce the decision rules and the leaf node is an outcome.

Classification trees are typically divided into two categories: Regression and classification trees. More often, they are used for classification problems. The classification tree algorithm not only provides the predicted class, but also the associated probability of belonging to that class [28]. As each method used for data processing, the basic classification tree algorithm

has its limitations. To overcome limitations such as overfitting or instability, the Random Forests or Gradient Boosting could be used.

#### Boosting and Bagging

Boosting is also a method of improving the prediction resulting from the decision tree. The algorithm creates multiple training sets of the original training dataset using bootstrap aggregation (also called bagging). On each of these sets, one boosting tree is grown sequentially. This means that each tree grows according to acquired information from previous trees. It is called the iterative method and a new model is influenced by the performance of previous models that were created before [28, 36].

### 2.4.4 Random Forests

Random Forest is a machine learning algorithm considered as a supervised learning method. It is used for both regression and classification problems and is based on the decision tree method. The higher the number of trees, the higher the accuracy and the lower the probability of overfitting [37]. In Random Forest, each sample $m$ is randomly chosen from the $p$ predictors. Typically, around $m \approx \sqrt{p}$ predictors predictors are taken in each split, as shown in Figure 2.10. This means that the algorithm does not consider all possible predictors. Consequently, when a very strong predictor occurs in the data, the algorithm compensates by only considering a subset of predictors in each split. On average, $(p - m)/p$ of the very strong predictor is not considered, allowing other predictors to contribute. The most important difference between bagging and Random Forest lies in the appropriate size chosen for $m$ [28].

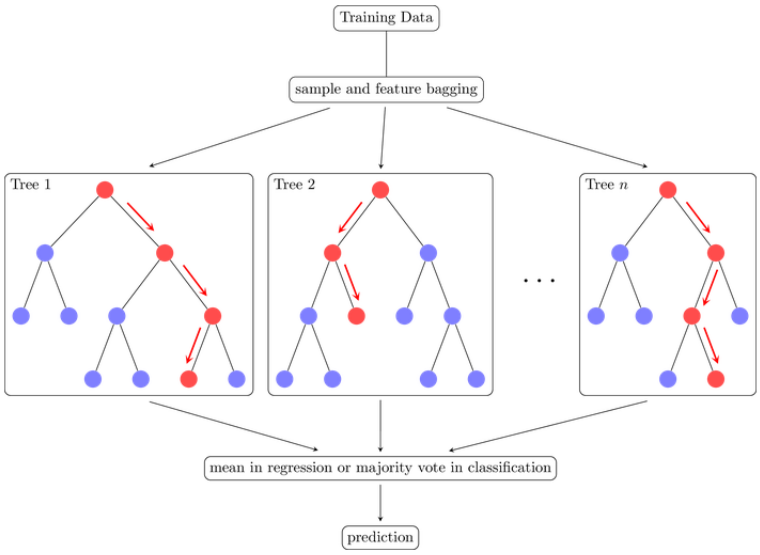



**Figure 2.10:** It describes the prediction of sample from dataset. At first, there is a train dataset. Then the trees are built associated with numbers of subsets. After that, the number of decision trees is chosen. In the figure Tree 1, Tree 2, ..., Tree n. These steps are repeated several times. Finally, for a new data point, the predictor of each tree (the last red leaf) is found. In regression, the predictor is chosen based on the mean, in classification based on majority votes.

### 2.4.5 Extreme Gradient Boosting

Extreme Gradient Boosting or XGBoost is an algorithm based on parallel tree boosting. The method was developed and specified in [38] with the idea of pushing the computing limits of

the boost algorithm to make it more robust. XGBoost creates new models that can predict the residuals of previous models. This means that new models learn from the difference between the observed and predicted data values. After the prediction of all residuals, the final prediction model is created [39]. The challenge of this algorithm is to choose the parameters for optimal performance as there are many of them. In this thesis, the set of parameters is found using the Grid Search algorithm.

The advantage of this method is its high speed and efficiency. Moreover, using in-build Lasso and Ridge regularization prevents overfitting and provides feature importance score to allow users to prioritize features that are more important for building the model. The disadvantage is the low interpretability of the model [39].

#### Grid Search Algorithm

Grid Search algorithm is a method to find the best suitable hyperparameters when tuning the model. The input is a set of values of different parameters. The algorithm calculates all possible possibilities and chooses the set of hyperparameters and their values that give the best results for the model after training. The disadvantage of the Grid Search algorithm is that it is time consuming and expensive. However, in a time-consuming and complexity trade-off, this method is the most suitable for the thesis according to the comparison in [40].

## 2.5 Model Evaluation Metrics

Evaluation methods are crucial to compare the predictive capabilities of models, fine-tuning model parameters, and assessing the ability of the statistical model to predict variables. This chapter introduces the evaluation metrics used to measure the performance of the models. Throughout the model training process, these metrics were calculated in the testing fold in each iteration of the cross-validation and averaged at the end to obtain the overall performance of the trained model.

### 2.5.1 Metrics for Regression

This section presents the evaluation metrics employed to assess the performance of regression models. These metrics provide insight into how well the models predict continuous variables.

#### R-squared

R-square method is a useful measure of the fitting quality for the regression models from the exponential family, which include, for example, logit, probit, Poisson or linear models [41]. It is defined as the proportion of variance in the dependent variable that is explained by the independent variables in a regression model, and mathematically it is expressed by equation 2.14.

$$R^2 = 1 - \frac{SSE}{SST}, \tag{2.14}$$

where SSE means squared sum of the error defined by equation 2.15 as

$$SSE = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \tag{2.15}$$

and SST is the squared sum of the data variance defined by equation 2.16 as

$$SST = \sum_{i=1}^{n} (y_i - \overline{y_i})^2 \tag{2.16}$$

and $y_i$ is the observed value of the dependent variable for data point $i$, $\hat{y}_i$ is the predicted value of the dependent variable for data point $i$ obtained from the regression model, $\overline{y_i}$ is the mean of the observed values of the dependent variable and $n$ is the number of data points [31].

If $R^2 = 0$ the model does not explain any of the variability in the dependent variable, if $R^2 = 1$ indicates that the model perfectly explains the variability in the dependent variable. However, the optimal $R^2$ results differ depending on the type of data and give any information about the significance of the variable. Therefore, it must be interpreted with other evaluation metrics.

#### Root Mean Squared Error

Root mean squared error or RMSE is a metric for evaluating the accuracy of the predictive model and provides an overall value of the error distribution. It expresses the average magnitude of the differences between predicted and observed values, providing a measure of how well the model predictions align with the actual data. This metric is defined by the equation 2.17 [42].

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}, \tag{2.17}$$

where the meaning of used variables is the same as by equations in Chapter 2.5.1. The best result is if $RMSE = 0$, which means perfect predictions.

#### Mean Absolute Error

Mean absolute error (MAE) is another accuracy evaluation metric for regression analysis. It represents the average absolute difference between the predicted and observed values, providing a measure of the model performance [42] and it is expressed with the equation 2.18 as

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|. \tag{2.18}$$

The meaning of used variables is the same as by equations in Chapter 2.5.1.

#### Effective Degrees of Freedom

Effective Degrees of Freedom or effective DoF is a measure that expresses flexibility and complexity of the model. It is useful for understanding the trade-off between model complexity and overfitting. This metric is present as the in-build criterion of the GAM summary table. The general rule is that the higher the effective DoF, the more flexible the model is to handle complex data, and there is a higher probability that the model will correctly fit the data and prevent overfitting [43].

### 2.5.2 Metrics for Classification

In this section, we present the metrics used to evaluate the performance of classification models. The evaluation metrics employed in this thesis are generalized for multi-class classification and are derived from the confusion matrix [44], as further described in this chapter. In binary classification, simplification involves excluding the variable $k$ from the equations.

### Confusion Matrix

The confusion matrix provides a useful visualization of a model performance. Typically, as shown in Figure 2.11, the x-axis represents the predicted classes of the classification model, while the y-axis displays the actual classification classes. From this table, all classification accuracy metrics can be derived. The variables in the matrix have the following meanings: TP denotes true positives, indicating the number of correctly predicted target instances; TN represents true negatives, indicating correctly predicted instances different from target ones. On the other hand, FP indicates the number of instances that were predicted as target class but actually belong to the other class, while FN represents instances that should have been predicted as target class but were instead assigned a different one [45]. Additionally, in the equations below, the variable $K$ is utilized to represent the total number of values in the confusion matrix.

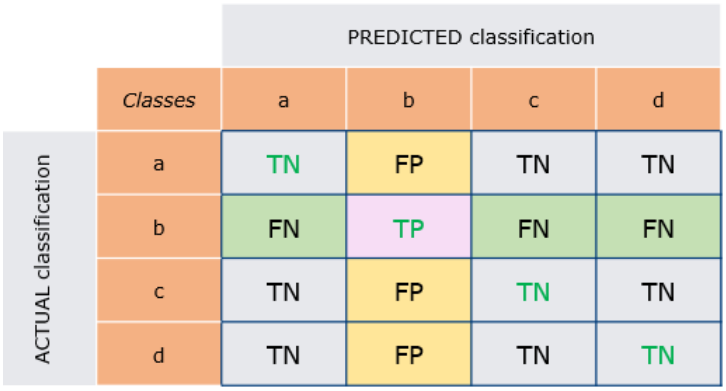



**Figure 2.11:** Confusion matrix for for multi-class classification. This matrix helps with understanding of evaluation techniques as all metrics are shown. In the figure, b is the target class [46].

### Accuracy

Accuracy measures the proportion of correctly classified instances divided by total number of instances in dataset.

$$Accuracy = \frac{\sum_{k=1}^{n} TP_k}{K} \tag{2.19}$$

### Precision

Precision measures the proportion of true positive predictions divided by all instances predicted as positive by the model. Indicates the model's ability to avoid false positives.

$$Precision_k = \frac{TP_k}{TP_k + FP_k} \tag{2.20}$$

### Recall

Recall measures the proportion of true positive predictions over all ground truth positive instances in the dataset. Indicates the model's ability to correctly identify all positive instances.

$$Recall_k = \frac{TP_k}{TP_k + FN_k} \tag{2.21}$$

### F-1 Score

The F1-score is the harmonic mean of precision and recall and provides a balanced measure of the performance of a model. It is particularly useful when dealing with inbalanced datasets.

$$F1_k = 2 \cdot \frac{Precision_k \cdot Recall_k}{Precision_k + Recall_k} \tag{2.22}$$

# Chapter 3

## Methods and Implementation

The implementation of selected methods has a pipeline that is shown in Figure 3.1. In the first step, feature engineering is performed where all implemented features are extracted from the measured EIT signal. After that, the clearing of created dataset is done followed by normalization after outlier detection. The next step keeps only the features that have a mutual correlation less than 85 %, and then they were eliminated based on their importance to the reference bolus signal. From these features, polynomial relationships of the second and third degree are found and ranked again according to their importance relative to the bolus signal. After the whole described process is completed, the final feature subset is created, and this is the input dataset for the implementation of statistical models.

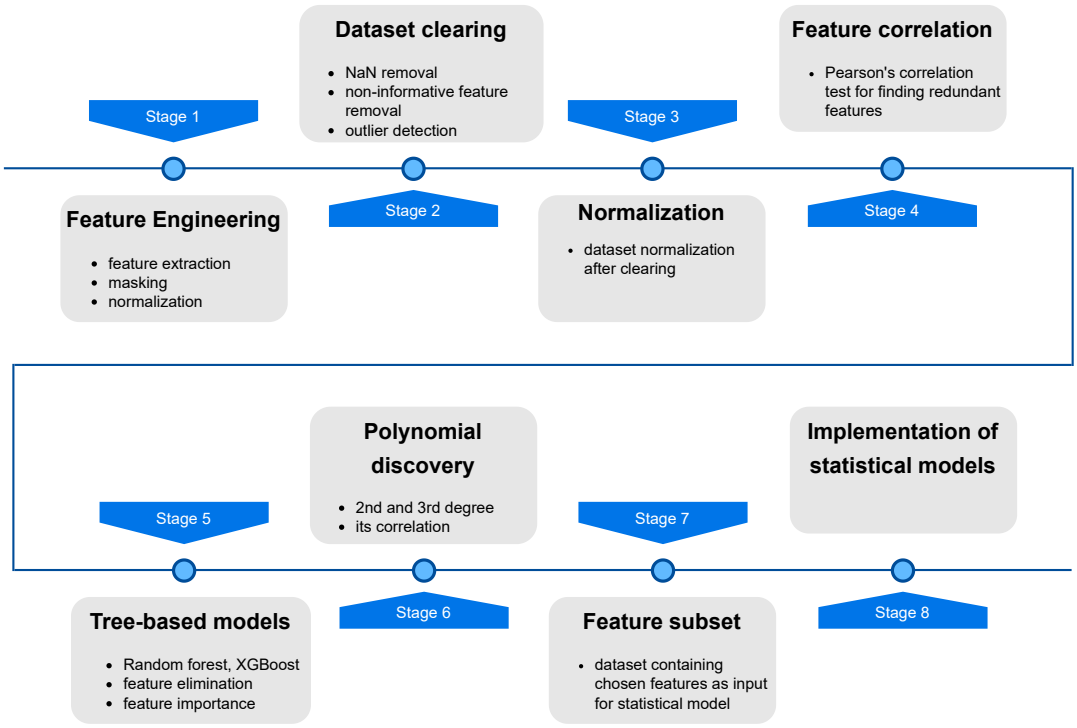



**Figure 3.1:** Implementation pipeline from dataset creation until inputs for statistical models.

## 3.1 Dataset Creation

This thesis utilizes recordings of EIT measurements of the field of view shown in Figure 3.2, which were initially processed using Matlab. To ensure accurate assessment of the lungs and heart, regions of interest (ROI) were extracted from CT measurements. The IBS flow served as a reference for the analysis.

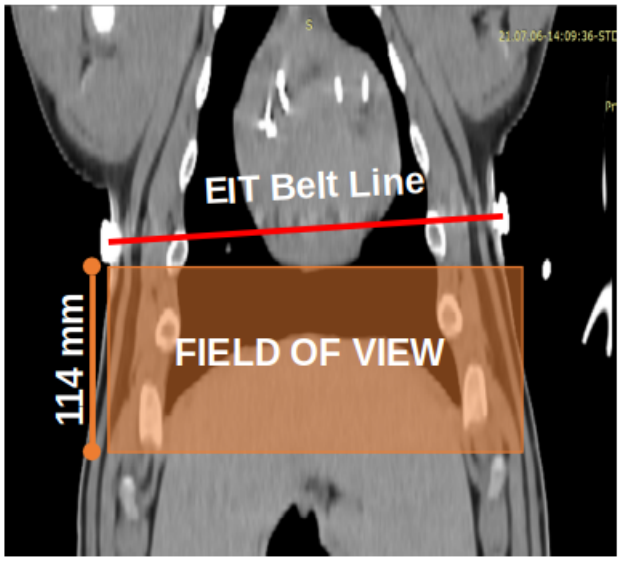



**Figure 3.2:** The placement of an EIT belt and field of view on real animal.

The dataset comprises measurements of 11 animals, recorded in multiple stages according to the blockage distance. The measurement of one stage takes approximately 18 minutes and was divided into several phases as shown in Figure 3.3. The first phase begins with apnea, during which respiration is paused, followed by bolus injection and measurement of IBS flow, labeled IBS 1. This entire phase lasts for one minute. The subsequent phase involves the EIT measurement, denoted CRS 1, conducted during the next half-minute. Afterwards, CT data is collected over 15 minutes. Finally, IBS 2 and CRS 2 are acquired again to complete the measurement of the entire stage. These procedures were followed to obtain all recordings.

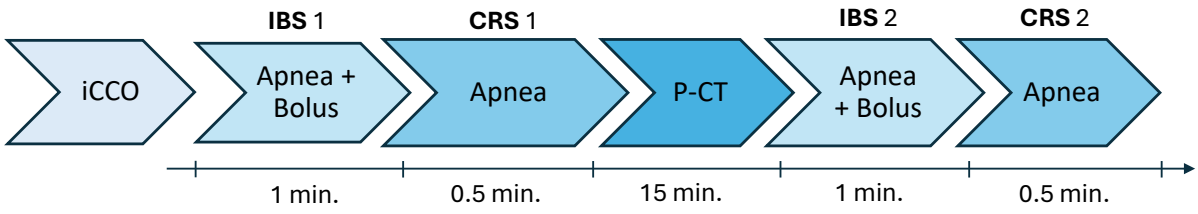



**Figure 3.3:** The process of data acquisition.

Some recordings show scenarios without blockage, while others capture instances of blockage, classified as central, medial, or peripheral, and marked as unhealthy. Central and medial blockages are similar as they both involve obstructing an artery from the heart to the lung, leading to reduced blood flow to the lung. Typically, central blockage occurs slightly before medial blockage. However, peripheral blockage occurs when the capillaries are obstructed, causing prolonged blood flow to the lungs before blockage. Blockage methods include catheterization and microsphere administration, with catheterization mainly resulting in

central or medial blockages, while microsphere administration leads solely to peripheral blockages.

### 3.1.1 Pre-processing

The dataset creation process begins with filtration in Matlab, where the cardiac signal was extracted from the overall signal using zero-phase digital filtering as proposed by Gustafsson [47]. Subsequently, the filtered data were processed in Python for subsequent statistical analysis.

Moreover, the CT signal played a crucial role in accurately determining the Regions of Interest (ROIs) for the lungs and heart. These ROIs were used in the model learning process, helping to mitigate the influence of perfusion not related to the lungs. Specifically, the ROIs were designed to precisely specify areas where lung signals occur, avoiding the middle part of the thorax, where signals from the heart and lungs are mixed, making it difficult to distinguish the organ responsible for the EIT signal. Individual components are depicted in Figure 3.4.

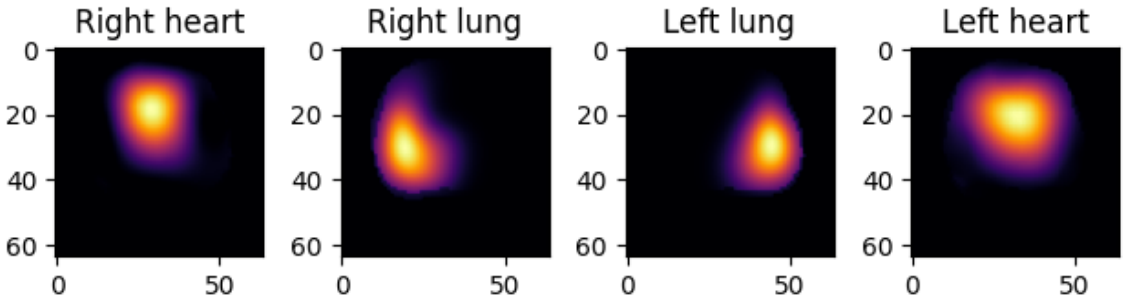



**Figure 3.4:** ROI importance map.

### 3.1.2 Label Extraction

In supervised learning, labels are a crucial part of the statistical learning process. They are references for models on which its predictions are evaluated, and models try to improve its performance in the next iteration. Correct label extraction helps statistical models in the learning process and significantly contributes to the training, evaluation, and prediction process. In regression task, the labels represent continuous values that the models try to predict based on given input features. The aim is to learn the relationship between input features and the corresponding labels to predict new data. However, for regression, no special algorithm for label extraction was developed.

In classification, labels represent distinct classes or categories. During the learning process, the model attempts to classify new data points based on input features and reference class labels. In this thesis, a binary classification is performed between healthy and unhealthy data points. Subsequently, this classification is extended to four classes, where the healthy and different types of unhealthy classes (central, medial and peripheral) are distinguished. The process of label extraction is shown in Algorithm 1.

Before starting the algorithm, masks for each lung (maskRL and maskLL) were established by selecting values from the signal in ROIs above a predefined threshold, Lungthres, which was set at 50% of the ROI signal for each lung. The IBSthres parameter was used to identify regions where the IBS flow reached at least 50% of its maximum amplitude. Within the algorithm, healthy or unhealthy labels were assigned to each pixel. During the healthy stage, the healthy label was assigned to both lungs. In unhealthy stages, the label was assigned

**Algorithm 1:** `LabelExtraction`

**Input:** maskRL, maskLL, $IBS_{thres}$, stages
**Output:** labels

labels = $\emptyset$;
**for** $s \in$ *stages* **do**
  **if** $s$ *is unhealthy* **then**
    $s_h$ = `find corresponding healthy stage to` $s$;
    **if** *isRightLung* **then**
      $IBS_{maxRL} \leftarrow$ `max from right healthy lung in` $s_h$;
      labels$_s$[maskRL$_s$ **and** IBS $< IBS_{maxRL} \cdot IBS_{thres}$] = unhealthy;
      labels$_s$[(maskRL$_s$ **and** IBS $>= IBS_{maxRL} \cdot IBS_{thres}$) **or** maskLL$_s$] = healthy;
    **else**
      $IBS_{maxLL} \leftarrow$ `max from left healthy lung in` $s_h$;
      labels$_s$[maskLL$_s$ **and** IBS $< IBS_{maxLL} \cdot IBS_{thres}$] = unhealthy;
      labels$_s$[(maskLL$_s$ **and** IBS $>= IBS_{maxLL} \cdot IBS_{thres}$) **or** maskRL$_s$] = healthy;
  **else**
    labels$_s$[maskRL$_s$] = healthy;
    labels$_s$[maskLL$_s$] = healthy;

based on the side, to the region where the IBS flow was low relative to the maximal flow in the corresponding healthy lung. Other pixels were designated as healthy since they remained unaffected by blockage. For the multi-class classification task, various types of unhealthy labels based on the stage were assigned to pixels. Details of both classification task labels are provided in Table 3.1.

| Task | Label | Meaning | Abbreviation |
|---|---|---|---|
| Binary classification | 0 | Healthy | **H** |
| | 1 | Unhealthy | **U** |
| Multi-class classification | 0 | Healthy | **H** |
| | 1 | Central blockage | **C** |
| | 2 | Medial blockage | **M** |
| | 3 | Peripheral blockage | **P** |

**Table 3.1:** Labels for both binary and classification tasks.

### 3.1.3 Feature Engineering

Following the pre-processing stage, feature extraction was performed on the acquired data. Pixel-wise features were extracted from the EIT measurements for each animal and stage over time, forming the dataset. Specifically, a pixel in the center of the heart is taken as a reference for signal delay computation and to align the reference pixel with other pixels to obtain the systolic and diastolic parts of the recording. In total, 107 features were created and extracted to support both the regression and classification tasks. More details on these features are provided in Table 3.2, with each feature being extracted for every pixel over time.

| Group | Feature | Description |
|---|---|---|
| Spectral | harmPeakX<br>harmIntX<br>harmDelX<br>harmPeakRatX<br>harmIntRatX<br>harmDelRatX | All of the spectral features are computed based on FFT spectrum of EIT recording over time. X represents the harmonics of the corresponding feature. harmPeakX is the amplitude of the Xth peak in the spectrum which is higher than 0.25 of maximum amplitude of the reference pixel over time spectrum. harmIntX sums up neighborhood amplitudes around the Xth peak in the spectrum. harmDelX is the spectral phase of the Xth peak. All Rat are ratios of the respective features to the total sum of those features over the whole EIT recording. |
| Optical | optDiastPha<br>optSystPha<br>optDiastR<br>optSystR | average Farnebäck optical flow [48] of Diastolic or Systolic part of EIT signal of each pixel in time transformed to polar coordinates (radius R and phase Pha) |
| Model-based | gaussXAY<br>gaussXmuY<br>gaussXsigmaY | sum of X = 2 or X = 3 Gaussians is fit to the EIT signal of each pixel in time and the sorted amplitudes A, means mu and standard deviations sigma of each Gaussian Y is one feature |
| Statistics | statsCumQX | quantile (corresponding percentiles X are $1 \rightarrow 2.5\%$, $2 \rightarrow 25\%$, $3 \rightarrow 50\%$, $4 \rightarrow 75\%$, $5 \rightarrow 97.5\%$) of the EIT signal of each pixel in time |
| | statsCumdQX | quantile differences divided by percentile difference $\frac{q_2-q_1}{p_2-p_1}$ |
| | statsMin | minimum of the EIT signal of each pixel in time |
| | statsMax | maximum of the EIT signal of each pixel in time |
| Temporal | tempLag | temporal lag with respect to the reference pixel calculated using searching for maximum correlation with the reference pixel |
| | tempEnergy | sum of squared EIT signal values of each pixel in time |
| Second-order | diffMin | minimal discrete difference between consecutive values of the EIT signal of each pixel in time |
| | diffSTD | standard deviation of the discrete difference of the EIT signal of each pixel in time |
| | diffXMin | time of the minimal discrete difference between consecutive values of the EIT signal of each pixel in time |
| | diffMedian | median discrete difference between consecutive values of the EIT signal of each pixel in time |
| | diffCentroid | centroid of discrete difference between consecutive values of the EIT signal of each pixel in time |
| Others | distToHeart | distance to the reference pixel of each pixel |

**Table 3.2:** Examples of extracted EIT features for both regression and classification tasks.

### Final Dataset

Before finalizing the dataset, standardization is applied. This process, also known as feature scaling, involves normalizing the range of features in a dataset. Features often have varying ranges due to differences in measurement quantities, thus standardization ensures consistency for statistical models to interpret them effectively. With normalization, each feature influences

| Animal | Stage | Record | px (x coord) | py (y coord) | features (107) | distance to heart | ROIs | No blocked | Central block | Medial block | Peripheral block | bolus |
|---|---|---|---|---|---|---|---|---|---|---|---|---|

**Figure 3.5:** Visualization of the structure of the final dataset that is used for training and testing of statistical model.

the model equally, which is also crucial for linear models to maintain interpretability, ensuring that all features hold the same relative importance. In this thesis, the Min-Max scaling was used, as it transforms the data into the same [0,1] range [49] and is expressed as

$$X_{\text{scaled}} = \frac{X - \min(X)}{\max(X) - \min(X)}, \tag{3.1}$$

where $X_{scaled}$ represents the scaled value of the feature $X$, $X$ is the original value, $X_{min}$ is the minimum value and $X_{max}$ is the maximum value [50] .

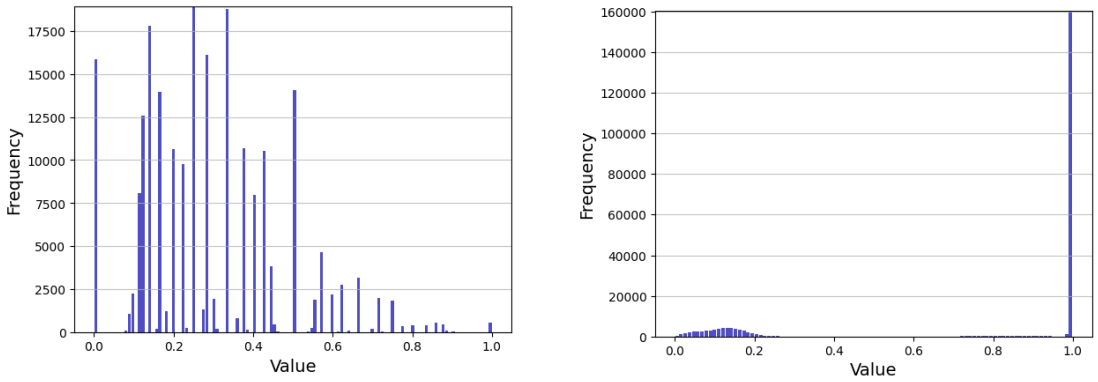
Once the standardization is completed, the final dataset can be visualized as a table referring to Figure 3.5 where each column belongs to the recordings, features, or IBS signal. Each row represents a single data point that represents a pixel of the EIT image. The final dataset was created for both the regression and classification tasks in a unified file.

### 3.1.4 Dataset Clearing

Data clearing is one of the most essential procedure in handling extensive datasets. It involves the elimination of incomplete, inaccurate, or irrelevant data from the dataset to improve its quality. It helps to eliminate erroneous data that could potentially mislead a model and ensures the creation of a high quality dataset [51].

In the initial step of this process, NaN values were removed, as they contain any value, to ensure the integrity of the dataset and to prevent potential issues during the training of statistical models. Furthermore, certain models cannot process datasets containing NaN values. These problematic data points were eliminated from both the EIT and the reference (IBS flow) datasets. Initially, NaN values were identified within the IBS signal. Subsequently, any rows containing NaN values were removed, not only from the IBS data itself but also from the corresponding EIT data, to ensure the same sizes of the data. This process was then repeated for the EIT data containing NaN features. As a result, the dataset was reduced by approximately 27.5%, as the images have a rectangular shape and include areas outside the field of view that do not contain the desired data.

The next step involves removing all features that have a non-informative character, see Figure 3.6a, or were saturated to one value, see Figure 3.6b, and do not contain relevant information for training process. Elimination was performed using histograms, and, based on the data distribution on the plot, non-informative and saturated data were removed.




**(a) :** Non-informative feature

**(b) :** Empty feature

**Figure 3.6:** Example of non-informative features. The frequency describes the number of pixels with the same value that is normalized in the range from 0 to 1.

### Outlier Detection

The final but important step in the clearning process of the dataset is the detection of outliers. It is an important part in improving the reliability of statistical analysis, preventing biased decisions, and crucial for preservation of the robustness of statistical models. According to the comparison [52] of anomaly detection algorithms in 2D datasets, outlier detection was performed using the isolation forest algorithm [53]. This method is effective in high-dimensional data and for large datasets.

The isolation forest, an outlier detection algorithm, was introduced in [53] and is based on random forests. The method randomly selects a subset of features and splits the data. Samples that traverse deeper into the tree are considered as those where outliers do not occur, as more cuts are needed to isolate them. Conversely, samples that are removed early on are deemed outliers as they are easier to separate. Recursive separation is determined by a tree structure, where the number of splits required to isolate the feature is indicated by the length from the root node to the terminating node. The advantage of this method is that it requires only a small number of trees and a small subsampling size to achieve high detection performance [53, 52]. To tune the algorithm, the values shown in Table 3.3 were set.

| | n_estimators | contaminationn | max_samples |
|---|---|---|---|
| Isolation Forest | 100 | 0.1 | auto |

**Table 3.3:** The best parameters for Isolation Forest algorithm.

Where contamination presents a number of anomalies over the total number of points. The isolation forest with the parameters mentioned above has the best performance for the purpose of this thesis. Data reduction decreased approximately 15. 6% in regression data and 11.8% for classification data.

### 3.1.5 Feature Correlation

The next step in data preparation for the statistical model is to remove features with high mutual correlation and also to find the most correlated features with the IBS signal, which is necessary for proper usage of linear models.

The phenomenon that occurs due to linearly correlated features used by training statistical model is called multicollinearity. It results in a situation where two or more independent

variables in a regression model are highly correlated, making it challenging for the model to separate and estimate the individual effects of each variable on the dependent variable. Avoiding this problem is an important part for creating and training reliable and interpretable regression models.

Initially, the problem of high mutual correlation among features was not addressed, features were used only based on Pearson's correlations with IBS flow, described in Chapter 4.1.1. Due to the discovery of multicollinearity, it was necessary to develop a new method that reflects this issue. The proposed method, detailed in Algorithm 2, incorporates both mutual correlations among features and the importance of these features to the IBS flow. Importances were computed using two methods: the XGBoost regressor and the Spearman correlation coefficient. The Spearman method proved to be more effective in feature selection, as discussed in Chapter 4.1.2, and also had a lower computational time compared to the XGBoost approach.

**Algorithm 2:** `FeatureCorrelation`

0.9
**Input:** features, $\text{corr}_{thres}$, IBS flow
**Output:** remaining features

$P \leftarrow$ compute pair-wise feature Pearson's corr;
**for** $f \in features$ **do**
  $imp \leftarrow$ find importances to IBS flow;
  $\max_{corr} = \max(P)$;
  $i, j \leftarrow$ indices of $\max_{corr}$ features ; // $i \neq j$
  **if** $max_{corr} > corr_{thres}$ **then**
    **if** *imp[i]* $<$ *imp[j]* **then**
      remove feature $i$ ; // also from $P$
    **else**
      remove feature $j$ ; // also from $P$
  **else**
    **break**;

## 3.2 Regression Task

Regression models estimate the relationship between a dependent variable (or outcome) and one or more independent variables, also known as features or inputs. These models play a crucial role in understanding data trends and modeling various scenarios. The goal of the regression problem is to estimate lung perfusion using a selected subset of features. The model is trained on this feature subset together with the reference IBS flow data, in order to accurately predict lung perfusion based on the features extracted from the EIT signal.

Following the completion of all steps outlined in the pipeline shown in Figure 3.1, regression models have been implemented. Due to the huge variety of statistical regression models, the task was to implement a small selection of them and compare their results. Figure 2.7 illustrates the trade-off between interpretability and performance among these models, where higher interpretability often corresponds to lower predictive performance and vice versa. As discussed in Calvaresi et al. [32], interpretability is a crucial aspect for humans to understand the results of statistical models.

There is a wide range of regression models available, with linear regression being the most common technique, serving as the baseline model against which the outcomes of more complex

models are compared. Generalized Additive Models (GAMs) and Extreme Gradient Boosting (XGBoost) models were chosen and implemented, alongside a Random Forest regression model for testing purposes. The mathematical background of these methods is detailed in Chapter 2.4. Here, only the implementation of these model is described.

The performance of predictive statistical models was evaluated using the cross-validation method, as described in detail in the following subchapter. The results of each model were evaluated using metrics such as R-squared, root mean squared error (RMSE), and mean absolute error (MAE), with explanations provided in Chapter 2.5.1.

### 3.2.1 Recursive Feature Elimination for Regression

Recursive Feature Elimination (RFE) is an algorithm designed to select the most relevant features from a training dataset to predict the target variable. It operates iteratively, initially training the estimator with all features and then ranking them based on their importance as described in Chapter 3.2.2. With each iteration, the algorithm eliminates the least important feature until the desired subset is achieved. This approach helps improve the performance of the model and prevent overfitting [54, 55].

In the initial stage, the features identified from Algorithm 2 were fed into the RFE algorithm to determine the most appropriate ones. The algorithm assigns weights to the features to achieve the desired number of selected features. These weights are determined by the estimator, a hyperparameter of the algorithm [56]. In this case, the estimator employed was the Random Forest Regressor with default parameters, with the objective of retaining 10 features.

### 3.2.2 Feature Importance

Following this, the features were ranked based on their significance to the IBS signal. The feature importance, represented as a real number, helps to determine which features contribute more significantly to the prediction process. One approach to reach the feature importances is through the random forest method. This method computes the overall importance of a feature by considering all tree splits where the feature is utilized, measured by reduced variance. It offers a multivariate feature importance score that is computationally efficient and applicable to multidimensional data [57, 58]. The total importances are scaled to 100, ensuring that each importance value reflects the fraction of the overall importance of the model [31]. Consequently, the relevant subset of features from the dataset is identified. The regression feature subset is plotted on the visualization grid, shown in Chapter 4.1.4.

The crucial aspect when implementing this algorithm is the selection of the estimator. Initially, the random forest regressor was used. However, its use proved to be more computationally and time-intensive compared to the XGBRegressor from the gradient boosting methods outlined in Chapter 2.4.5, with the resulting rankings being nearly identical. Consequently, the XGBRegressor was selected for its efficiency.

### 3.2.3 Cross-Validation

The cross-validation (CV) method is a technique for evaluating statistical model performance, enabling comparison, tuning, and selection of the most suitable model for specific research purposes. In this thesis, the Leave-One-Subject-Out Cross-Validation (LOSOCV) principle was chosen and is described below.

LOSOCV involves splitting the dataset into two parts: the validation set, consisting of a single fold $(x_1, y_1)$ representing one subject (or animal in this case), and the training set comprising the remaining observations $(x_2, y_2), ..., (x_n, y_n)$. The model is then trained on

$n-1$ training folds, and a prediction $\hat{y_1}$ is made for the value $x_1$. The unbiased test error $MSE_1 = (y_1 - \hat{y_1})$ is calculated as a result of this single fit process [59, 28]. However, relying only on this result for model evaluation is inadequate due to its high variability from acquiring a score from only one observation. To address this, LOSOCV repeats the process with each $(x_i, y_i)$ pair for validation and the remaining $n-1$ observations for training. By iterating this process $n$ times, the test mean squared error (MSE) is computed as the average of all test errors as [28]

$$MSE_{CV} = \frac{1}{n}\sum_{i=1}^{n} MSE_i. \tag{3.2}$$

As mentioned above, each observation corresponds to a single animal, and the LOSOCV algorithm is executed 10 times to derive the final MSE. It was observed that the LOSOCV MSE for each animal, denoted as $MSE_i$, varied slightly. This variation can be attributed to the uniqueness of each animal and potential differences in conditions during the EIT measurement.

An advantage of the LOSOCV principle is that the data points of one animal do not influence both the training and testing processes. In contrast, with a simple CV, which randomly selects data points without considering animal dependency, the results may be biased due to inherent similarities among data points from the same animal.

### 3.2.4 Implementation of GAMs

For implementation of GAMs the `pygam()` library was used [35]. As a first step, the LinearGAM and ExpectileGAM models were imported. From [35] it is known that LinearGAM has identity link and normal distribution, ExpectileGAM also has identity link with normal distribution but minimizing the Least Asymmetrically Weighted Squares.

All models were examined, and their results were compared. This comparison was based on the Grid Search results using the generalized cross-validation (GCV) score with the parameters outlined in Table 3.4. The resulting parameters are detailed in Chapter 4.2.1. The hyperparameter lambda controls the amount of smoothing applied to the spline basis function, influencing the complexity of models and mitigating overfitting. Smaller lambdas afford greater model flexibility, while larger lambdas help prevent overfitting. Splines, which are non-linear functions, help in modeling the relationship between predictor and response variables. Unlike lambdas, splines are specific parameters of the LinearGAM model. Expectiles, on the other hand, do not directly impact smoothing or model complexity, but facilitate accommodating non-normality or heteroscedasticity in the data. Lambdas are applicable across all models, while splines are a parameter solely of the LinearGAM model, and expectiles are applied only to the ExpectileGAM model.

| Hyperparameter | Values |
|---|---|
| Lambdas | 0.001, 0.00464, 0.0215, 0.1, 0.464, 2.15, 10, 46.4, 215.44, 1000 |
| Splines | 4, 6, 8, 11, 13, 15, 18, 20, 22, 25 |
| Expectiles | 0.05, 0.25, 0.5, 0.75, 0.95 |

**Table 3.4:** Parameters for GAM for Grid Search method.

The output of the GCV is a summary table displaying the number of degrees of freedom (DoF), AIC or pseudo R-squared metric. However, the last two parameters have only an informative character for a particular model. The decision was made according to the self-implemented LOSOCV for unbiased estimation of the model evaluation metrics.

### 3.2.5 Implementation of XGBoost

The XGBoost model, which is considered a non-interpretable model, was used based on [38]. Although the theoretical foundations are outlined in Chapter 2.4, this section focuses mainly on the practical application of this algorithm. XGBoost operates on tree-based principles, implying that only the following parameters mentioned in Table 4.2 were used to improve model performance, while others remained at their default settings.

The parameters have the following meanings. The max_depth parameter determines the maximum depth of the trees in the model, influencing both the model complexity and the computational resources; setting this value too high can lead to overfitting. The parameter n_estimators indicates the number of trees, with increasing values generally improving outcomes; however, similar to the maximum depth, it also carries the risk of overfitting. A typical range for this parameter is between 100 and 1000 [60]. Gamma represents the minimum loss reduction required to further partition the leaf node of the tree and contributes to regularization. A higher value corresponds to stronger regularization [38].

## 3.3 Classification Task

Classification models are designed to identify patterns between input features and categorical labels derived from a dataset. To address this problem, two approaches were taken. Initially, the classification was focused on distinguishing between healthy and unhealthy states, which was a binary classification. Once satisfactory results were achieved, the classification was extended to include three classes of unhealthy data points based on the location of the blockage, namely, central, medial, and peripheral.

The implementation pipeline is nearly identical as it is shown in Figure 3.1 and as it was employed for regression. However, two alterations were made. The first is in stage 5, where only RFE was used for feature selection. The reason is that there is no correlation between features and numerical labels, as there is for regression between features and the IBS signal. The correlation relationship with labels is adequately replaced by RFE. The second alternation relates to Stage 6, which was omitted since it is not applicable to the task.

In classification tasks, as in regression, there are various techniques available to create suitable models. Highly interpretable models such as Generalized Additive Models (GAMs) and Linear Regression, alongside more complex ones like Random Forest and XGBoost, were considered. The mechanisms of these models are detailed in Chapter 2.4. In the rest of this section, only their implementation and parameter usage are discussed for the purposes of this thesis. The implementation details of all of these models are discussed in this chapter. To ensure independent evaluation of the classification models, cross-validation was employed. Similarly to regression, the LOSOCV approach was used, which is explained in Chapter 3.2.3. However, the evaluation metrics differ from those used in regression. Here, precision, recall, F1-score, accuracy, and confusion matrix are used for model performance evaluation. The meaning and interpretation of each metric are described in Chapter 2.5.2.

### 3.3.1 Recursive Feature Elimination for Classification

In the introduction to this chapter, it was mentioned that only the elimination algorithm was adjusted to determine important features of the classification. Two estimators, namely Random Forest and Logistic Regression, were applied in this process. The selected features, which were chosen as the most important, were similar for both estimators. However, for the final feature set, the features obtained from the Random Forest estimator were used to

train the classification model, and the feature subset is shown in Chapter 4.1.4. This decision was made because the Random Forest estimator is more general and capable of handling non-linear relationships, whereas logistic regression only identifies linear relationships.

### 3.3.2 Binary Classification

Binary classification is a supervised learning technique, in which a classifier is tasked to categorize input data into one of two categories. Each data point is assigned to a category, and the model learns to distinguish the boundary between them to achieve an accurate separation [31].

In this thesis, the classes are labeled as 0 for "healthy", further marked as **H** and 1 for "unhealthy", further as **U**. The goal is to correctly classify as many unhealthy data points as possible, as identifying blockages in the lungs is crucial.

This problem was addressed first to determine if it is possible to develop a model capable of distinguishing between different types of unhealthy classes. Given that there are significantly more healthy data points than unhealthy ones, see Table 3.5.

| Class | n. of data points | percentage [%] |
|---|---|---|
| **H** | 72116 | 81.4 |
| **U** | 9469 | 18.6 |

**Table 3.5:** The number of data points in each class for binary classification.

The assumption that dataset is imbalanced, which was detected in the label creation described in 3.1.2, was confirmed. Further in this chapter, the implementation of weighting and down-samplig to balance the dataset is introduced. The results of binary classification are presented in Chapter 4.3.2.

### 3.3.3 Multi-class Classification

Multi-class classification, alike binary classification, is a form of supervised learning where each instance is associated with a single class label. The goal of classification models is to compute the probabilities of data points belonging to each class and assign them the highest probability [44]. In this type of classification, classes can be represented by numerical values (e.g., 0-9), sentiment categories (e.g. "positive," "ne.g.ative" or "neutral"), or object types (e.g. car, person, tree).

The categories were represented by numerical values: 0 for "healthy" (abbreviated as **H**), 1 for "central blockage" (abbreviated as **C**), 2 for "medial blockage" (abbreviated as **M**), and 3 for "peripheral blockage" (abbreviated as **P**). Here again, the goal is to accurately classify the majority of data points. As is known from binary classification, the dataset is imbalanced. The distribution of data points in each category is shown in Table 3.6.

| Class | n. of data points | percentage [%] |
|---|---|---|
| **H** | 72116 | 81.4 |
| **C** | 4468 | 5.1 |
| **M** | 2521 | 2.8 |
| **P** | 9469 | 10.7 |

**Table 3.6:** The number of data points in each class for multi-class classification.

Because of the significant difference in the number of data points in each class, weighting and down-sampling techniques were utilized to address dataset imbalance and mitigate the risk of overfitting on a single class, specifically the healthy class. The results of the multi-class classification are elaborated in Chapter 4.3.3.

### 3.3.4 Weighting

Weighting is a technique that is beneficial for imbalanced datasets, particularly when one class has a significantly larger number of samples than the others. The approach involves assigning higher weights to the minority classes, which enables the model to focus more on the accurate classification of these data points and helps prevent bias toward the majority class [61]. This issue, as previously mentioned, also occurred in the dataset analyzed in this thesis. Weights were initially applied using the formula in equation 3.3 for binary classification and equation 3.4 for multi-class classification as

$$\begin{aligned} H : w_0 &= \frac{n_{samples}}{2 \cdot (n_{samples} - n_{unhealthy})} \\ U : w_1 &= \frac{n_{samples}}{2 \cdot n_{unhealthy}} \end{aligned} \tag{3.3}$$

$$\begin{aligned} H : w_0 &= \frac{n_{samples}}{4 \cdot n_H} \\ C : w_1 &= \frac{n_{samples}}{4 \cdot n_C} \\ M : w_2 &= \frac{n_{samples}}{4 \cdot n_M} \\ P : w_3 &= \frac{n_{samples}}{4 \cdot n_P}. \end{aligned} \tag{3.4}$$

More about weighting is described in Chapter 4.3 as the weights were adjusted according to the results of model.

### 3.3.5 Downsampling

Another efficient technique for working with imbalanced datasets is down- or oversampling of data. For the purpose of the thesis and preservation of the pure real dataset, only downsampling of dataset was implemented. Moreover, by oversampling there is a high risk of overfitting and it takes more time to train the statistical model [46].

Downsampling is the process of randomly selecting data points from the majority class to create a subset of instances that is equivalent in size to the minority class. However, this approach also has its drawbacks, as it can result in the loss of potentially valuable data [62].

This approach was utilized due to poor results after dataset weighting. Even though balancing the classes through weighting was attempted, the models continued to overfit the predominant class. In scenarios involving multi-class classification, the predominant class **H** was downsampled to match the combined total of the other three classes. This decision was made to prevent a significant loss of data points that would occur if it was downsampled to to match the number of data points of the least **M**. To further balance the dataset, weighting was implemented on the downsampled data to ensure that each class was given equal importance.

### 3.3.6 Implementation of GAMs

The GAM model was applied using the `pygam` library [35]. In the context of classification, the initial model tested was the Logistic model, which was imported as LogisticGAM. The LogisticGAM uses a binomial error distribution and a logit link function. These parameters were left unchanged. To optimize the model parameters, the Grid Search method from `pygam` was employed, similar to the approach used in regression tasks. The results are presented in Chapter 4.3.

The model was evaluated using LOSOCV. To address the imbalanced dataset, weights were tried to incorporate. However, due to an implementation error of the library, the `pygam` did not accept them, as it resulted in an error. Due to the inability to adjust the data weighting in LogisticGAM, the results for the unhealthy class were poor, as there are significantly more healthy data points than unhealthy ones. LogisticGAM was also applied to the downsampled dataset without requiring dataset weighting. This is further elaborated on in the results in Chapter 4.3.

### 3.3.7 Implementation of Logistic Regression

Another interpretable model used was logistic regression, implemented using the scikit-learn library [63]. This model is appropriate for both binary and multi-class classification tasks. Various solver types are available, some designed only for a specific classification type. Nevertheless, there are solvers that can be used for both types, and this was utilized in the thesis. The penalty is determined by the solver, and these two parameters are by default linked. Although it is possible to adjust these parameters independently, not all solvers are compatible with all types of penalties. As a next parameter, the number of iterations was fine-tuned, which significantly impacts the model's performance. A low number of iterations may cause that the optimization algorithm does not reach the optimal solution, whereas increasing the iterations could enhance performance. In contrast, an excessively high number of iterations can result in overfitting and increased computational costs, particularly with large datasets. Parameters that are not specified here were set to their default values. Taking into account all described problems, the parameters shown in Table 4.6 were adjusted in the model tuning process.

### 3.3.8 Implementation of Random Forests

As an alternative example of the non-interpretable category, the Random Forest classifier was used with the scikit-learn library [64]. The reason behind selecting this model lies in its benefits, such as achieving high accuracy, avoiding overfitting, and providing interpretability through methods that display the impact of each feature on the model's performance. These characteristics make this model suitable for various classification tasks.

A Random Forest method was employed to address both binary and multi-class classification problems, as it is suitable for both scenarios. In the thesis, two hyperparameters, the number of estimators and the maximum depth were fine-tuned. The number of estimators corresponds to the number of trees in the forest. It is commonly observed that a higher number of estimators leads to better model performance. However, with an increasing number of estimators, the computational time also increases, and not every increment necessarily results in enhanced performance. There is also a boundary behind which it stops improving. The max depth parameter controls the depth of the trees, preventing the model from becoming too complex. This parameter helps to achieve a balance between the variance and bias trade-off and can also decrease the computational efficiency. The optimal values for each parameter were determined

using the Grid Search algorithm and are listed in Table 4.7. Other parameters of the model were set as default since they were most suitable for the dataset used.

### 3.3.9 Implementation of XGBoost Classifier

The last model used for classification is the XGBoost Classifier. It works similarly to Random Forest, but is typically less interpretable, especially when dealing with deep trees. One distinction lies in the way how trees are constructed. While Random Forest builds trees simultaneously, XGBoost does it sequentially, leading to increased model complexity. Nevertheless, XGBoost is frequently considered as the state-of-the-art method in classification tasks.

The XGB model was implemented using the XGB library [38]. This model offers parameter tuning to optimize performance on specific datasets. In this particular implementation, adjustments were made to the maximum depth, the number of estimators, and the learning rate. These parameters have the same interpretation as in Random Forest, with the exception of the learning rate, which was not defined in the Chapter 3.3.8. The eta parameter, also known as the learning rate, controls the step size by which the weights in the optimizer are updated. Selecting an appropriate value involves a trade-off, as a higher eta results in less accurate updates, but requires less time, while a lower value has the opposite effect [60]. The parameters were configured using the Grid Search algorithm and are shown in Table 4.7. Furthermore, the model enables the utilization of weights for both binary and multi-class classification classification.

## 3.4 Model Testing and Baseline Comparison

After the learning process, each model was validated and evaluated using LOSOCV to mitigate biased results.

To conclude the model evaluation, a comparison was made between the implemented model and a basic model known as the baseline model. The baseline model is a straightforward linear model fitted on a single feature. This feature is chosen based on its highest correlation or importance with the IBS flow. The specific feature utilized varies depending on whether the model is used for regression or classification tasks. For regression, the model is characterized by the feature diffMin, while for classification tasks (both binary and multi-class), it is constructed using the feature statsMin.

To ensure an unbiased comparison between the final model and the baseline model, an initial step involved processing the thirteenth animal separately for testing purposes. From this dataset, only NaNs were removed because the models were unable to handle data points containing NaNs and for proper visualization of the results. However, outliers were retained, as real-world data may indeed contain outliers. It should be noted that although the testing dataset was not extensive, it still produced unbiased results and that the volume of data was comparable to that of one animal in the LOSOCV evaluation. Subsequently, whenever results are derived from this testing animal, it is noted. Visualizations are done only for this testing animal.

# Chapter 4

# Results and Discussion

This chapter presents the results of consecutive stages shown in Figure 3.1 that lead to the final evaluation of the classification and regression tasks. The main focus is on the comparison of algorithms and models implemented under different conditions. First, the results of the pipeline are shown, which are necessary for creating the feature subset that serves as input for the statistical models. Then, the results of the regression task are presented, followed by the evaluation of the outcomes of the classification models. In addition, the training process, preparation of the testing data and the baseline model are explained.

## 4.1 Feature Selection

In the feature engineering process, the mutual correlation between features and the correlation of features with the IBS flow signal were examined. To choose the most suitable features, the Algorithm 2 was implemented. The results are given in the following sections.

### 4.1.1 Pearson's Correlation Test

To discover the correlation relationships between the IBS flow signal and the features, the Pearson correlation test was used. In Figure 4.1 the result of this algorithm is shown. It is necessary to mention that 107 features were implemented; some of them were removed due to their non-informative character as described in 3.1.4. In Figure 4.1, only the features that have more than 20% correlation with IBS flow are plotted.

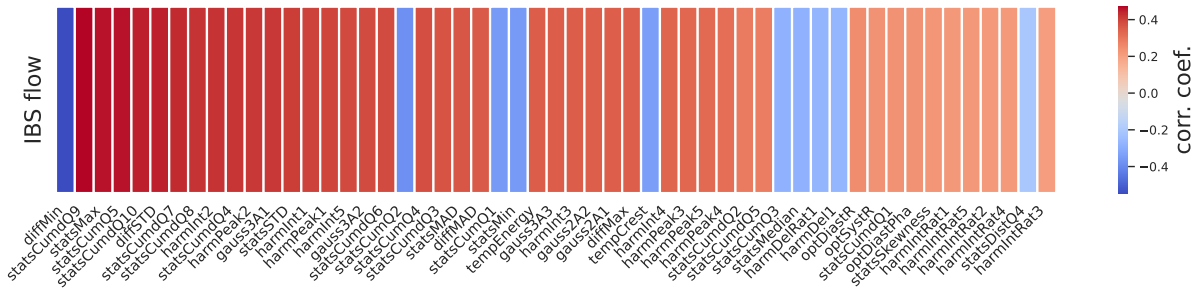



**Figure 4.1:** Feature correlation with IBS flow.

### 4.1.2 Results of Implemented Algorithm

This section presents the results of the self-implemented correlation test (Algorithm 2), which removed highly mutually correlated features based on their importance to the IBS flow using

either Spearman or XGBoost. Subsequently, the RFE method was employed to eliminate the remaining features that are considered unimportant with respect to the IBS flow. Table 4.1 displays the best features resulting from RFE, sorted according to their importance.

Based on Table 4.1, the features derived using XGBoost as the importance estimator are similar to those obtained using Spearman correlation coefficients as the decision-making method in Algorithm 2. While XGBoost took approximately an hour for computation, Spearman correlation reduced the computation time to several minutes. Therefore, Spearman correlation was selected as the decision method in Algorithm 2.

| Ranking | XGBoost | Spearman |
|---|---|---|
| 1 | diffMin | diffMin |
| 2 | harmDel1 | harmDel1 |
| 3 | distToHeart | distToHeart |
| 4 | optSystPha | optSystPha |
| 5 | statsCumdQ2 | optDiastPha |
| 6 | optDiastPha | harmDel2 |
| 7 | harmDel2 | harmDelRat4 |
| 8 | harmDelRat4 | statsCumdQ5 |
| 9 | harmInt5 | statsCumQ3 |
| 10 | harmPeakRat2 | harmInt5 |

**Table 4.1:** The ranking of importances for each correlation estimator.

Subsequently, a correlation graph between the IBS flow and the features obtained from the correlation test algorithm was generated (illustrated in Figure 4.2). Individual Pearson correlation coefficients were calculated and ranked to compare with the ranking of all the features shown in Figure 4.1. It became apparent that the implementation of an elimination algorithm to select uncorrelated features was necessary. Despite the initial features appeared to be correlated with the IBS flow (as shown in Figure 4.1), only diffMin and harmInt5 remained significant after the feature selection process (as shown in Figure 4.2). This underscores the importance of feature selection in identifying truly relevant features among potentially correlated variables.

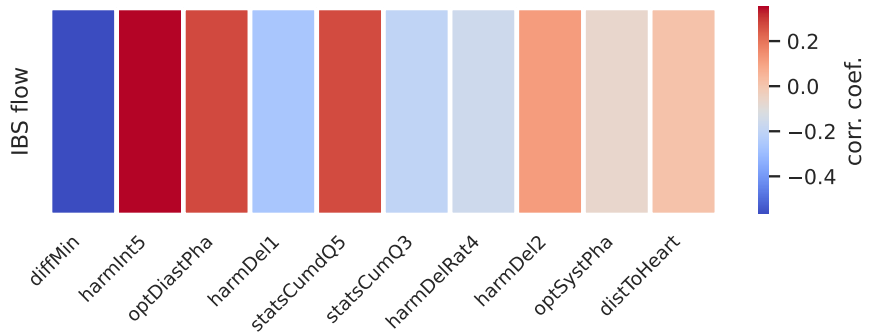


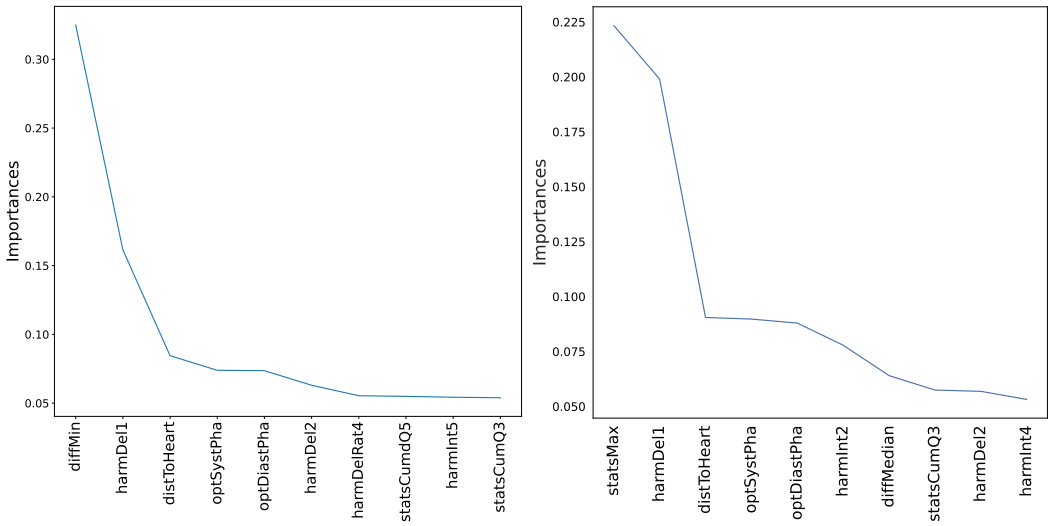

**Figure 4.2:** Correlation graph between IBS flow and features resulting from self-implemented Algorithm 2 using Pearson's correlation as as method for importance computation.

### 4.1.3 Feature Elimination

Model overfitting, a common issue during model training, can result in poor performance. It arises when a large number of features are included in the training process, making

it challenging to discern the relevant ones. Hence, the elimination of non-informative and irrelevant features becomes necessary. Initially, the RFE algorithm was configured to return 10 features, based on the number of features considered important by the correlation Algorithm 2. To address this, a graph (in Figure 4.3) depicting feature importances computed by the RFE algorithm was generated to determine the optimal number of significant features.

Past a certain point, the curve flattened, and features beyond this threshold were excluded from consideration for the statistical model. In this case, the cutoff occurred after the harmDelRat4 feature, as no change in the curve was noted beyond this point. Additionally, certain features were excluded for each task, despite their high correlation with the IBS flow signal or significant importance. One reason for this is that some features may accurately represent healthy lung areas, but do not adequately represent unhealthy regions. Since healthy areas outnumber unhealthy ones, this could bias the importance results. Consequently, the model relies heavily on these particular features, neglecting the potential contributions of others that may better represent obstructed areas in the learning process.




**(a) :** Before diffMin removel. **(b) :** After diffMin removal.

**Figure 4.3:** The importance curve after running the RFE algorithm before and after removing the diffMin feature.

During the model improvement process, it was observed that the diffMin feature is too dominant over healthy areas, causing other features to be suppressed. After removing the diffMin feature, the curve of the RFE algorithm is shown in Figure 4.3b, and illustrates that features beyond statsCumQ3 were cut off. Comparable plots were generated, determining a reasonable cut-off point for each task and the subtask of regression and classification. Features before the cut-off point were utilized in their respective tasks to train the statistical model.

### 4.1.4 Feature Subsets

The feature subset consists of the selected features for their respective classification or regression task. Figures 4.4 and 4.5 visually assess whether the previously mentioned outlier detection, feature elimination, and feature correlation algorithms effectively removed outlier data points, non-informative features, and highly correlated features. The plots confirm the validity of the assumptions. Notably, a 65% similarity between the statsMax and harmInt2 features in Figure 4.4 indicates their highest correlation. Despite this, these features were retained, as both features are considered important according to the graph in Figure 4.3b.

Additionally, if outliers occurred, they could be recognized from the point density below the diagonal. On the diagonal, the distribution of feature points is shown.

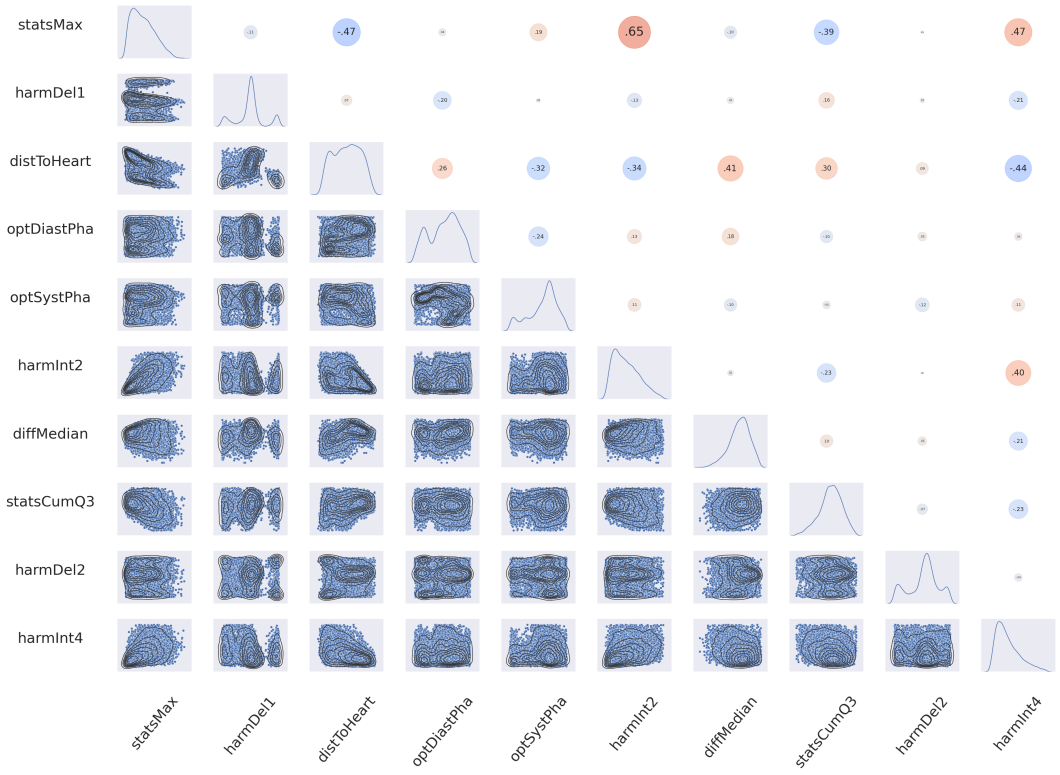



**Figure 4.4:** Visualisation grid for regression showing mutual correlations above diagonal, features dependency below diagonal and feature distribution density on diagonal.

## 4.2 Regression

This section presents an overview of the results obtained from the regression task. It begins with a discussion on the parameters of each model, followed by an evaluation of the models using their optimal parameters. The comparison between models is based on numerical results, scatter plots, and visualizations. Finally, the results from the testing animal are compared with those from the baseline model.

### 4.2.1 Hyperparameter Value Selection

This section shows the result of the Grid Search algorithm in Table 4.2 that is able to find the best hyperparameter performance values out of the initial set for all regression models implemented.

| Model | lambda | splines | expectile | max_depth | n_estimators | gamma |
|---|---|---|---|---|---|---|
| LinearGAM | 2.1544 | 25 | N/A | N/A | N/A | N/A |
| ExpectileGAM | 0.4642 | N/A | 0.5 | N/A | N/A | N/A |
| XGBoost regressor | N/A | N/A | N/A | 5 | 500 | 0.5 |

**Table 4.2:** The best parameters for each regression model according to the Grid Search.

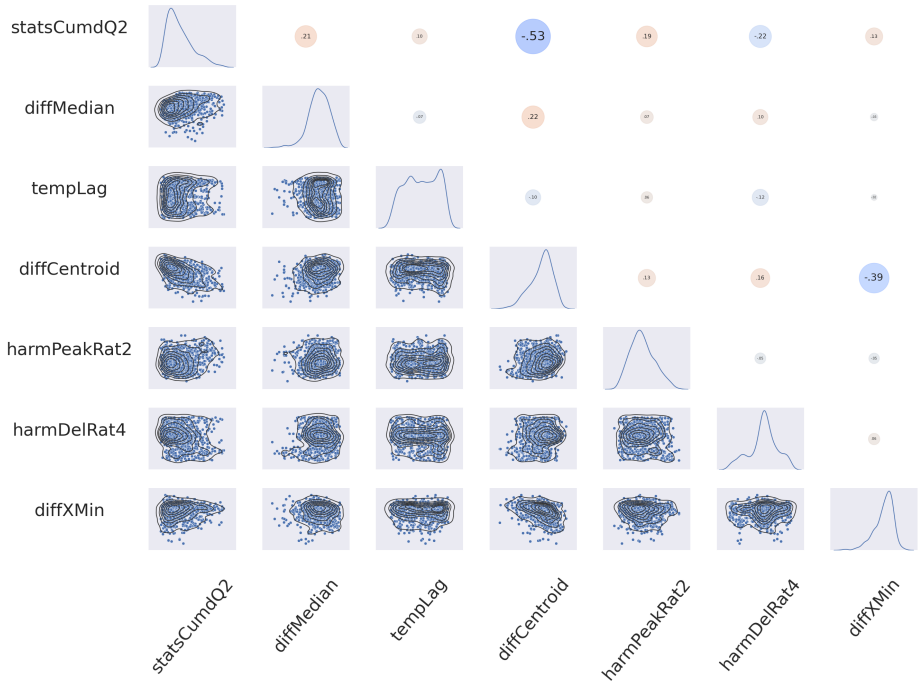



**Figure 4.5:** Visualisation grid for classification showing mutual correlations above diagonal, features dependency below diagonal and feature distribution density on diagonal. This feature subset represents chosen features for binary classification task.

### 4.2.2 Numerical Metrics

In this thesis, the metrics presented in Tables 4.3 and 4.4 were considered, and the results were evaluated using LOSOCV.

Upon comparison of the results, slight differences were observed in the evaluation metrics. Models trained without the diffMin feature exhibited slightly improved performance across all metrics and demonstrated a lower standard deviation. Consequently, the diffMin feature was excluded from the final set of features. However, given its significant correlation with IBS flow, as noted earlier, the baseline model was constructed based on this feature for comparison purposes.

According to the results presented in Table 4.3, although there are minor differences in effective Degrees of Freedom (DoF) between LinearGAM and ExpectileGAM, the chosen model for further evaluation and comparison is LinearGAM. The advantage of the LinearGAM model lies in its interpretability and comprehensibility compared to ExpectileGAM [31]. The importance of the effective DoF metric is described in Chapter 2.5.1.

| Model | Eff. DoF | AIC | Ps. $R^2$ | $R^2$ | RMSE | MAE |
|---|---|---|---|---|---|---|
| **LinearGAM** | 222.33 | $6.57 \cdot 10^6$ | 0.54 | 0.43±0.13 | 0.18±0.02 | 0.14±0.02 |
| ExpectileGAM | 228.51 | $6.57 \cdot 10^6$ | 0.54 | 0.41±0.13 | 0.18±0.02 | 0.15±0.02 |
| XGBoost | N/A | N/A | N/A | **0.47±0.12** | **0.17±0.02** | **0.12±0.02** |
| Baseline | N/A | N/A | N/A | 0.02±0.20 | 0.20±0.02 | 0.16±0.02 |

**Table 4.3:** Summary and evalution metrics for GAMs regression models without diffMin.

When comparing the results of the LinearGAM and XGBoost models, XGBoost exhibits better performance, particularly evident in the $R^2$ metric. This outcome was expected, given that XGBoost is a less interpretable model, which aligns with the performance-interpretability

| Model | Eff. DoF | AIC | Ps. $R^2$ | $R^2$ | RMSE | MAE |
|---|---|---|---|---|---|---|
| **LinearGAM** | 222.33 | $6.57 \cdot 10^6$ | 0.54 | 0.39±0.15 | 0.19±0.02 | 0.15±0.02 |
| ExpectileGAM | 228.51 | $6.57 \cdot 10^6$ | 0.54 | 0.40±0.15 | 0.19±0.02 | 0.15±0.02 |
| XGBoost | N/A | N/A | N/A | **0.47±0.16** | **0.17±0.03** | **0.14±0.02** |
| Baseline | N/A | N/A | N/A | 0.02±0.20 | 0.20±0.02 | 0.16±0.02 |

**Table 4.4:** Summary and evalution metrics for GAMs regression models with diffMin.

trade-off discussed in Chapter 2.4. However, since the objective is to find an interpretable model, LinearGAM is selected for further evaluation metrics.

The $R^2$ metric quantifies the proportion of variance in the dependent variable explained by the independent variable. According to Table 4.3, this value is 43% for LinearGAM. Although this value may not be considered high in a general context, the interpretation of what a high value of $R^2$ can mean varies depending on the application and the type of input data, as this statistical measure is largely influenced by the variation in the data [65]. Therefore, the state-of-the-art method and other evaluation metrics are needed when the $R^2$ metric is evaluated.

As illustrated in Table 4.3, the baseline model yields a notably low value $R^2$. Furthermore, the baseline model has a standard deviation of 0.2, indicating significant variability in the model outcomes, which is consistent across all models. This variability issue was further explored, and the metrics for each animal were visualized for LinearGAM in Figure 4.6.

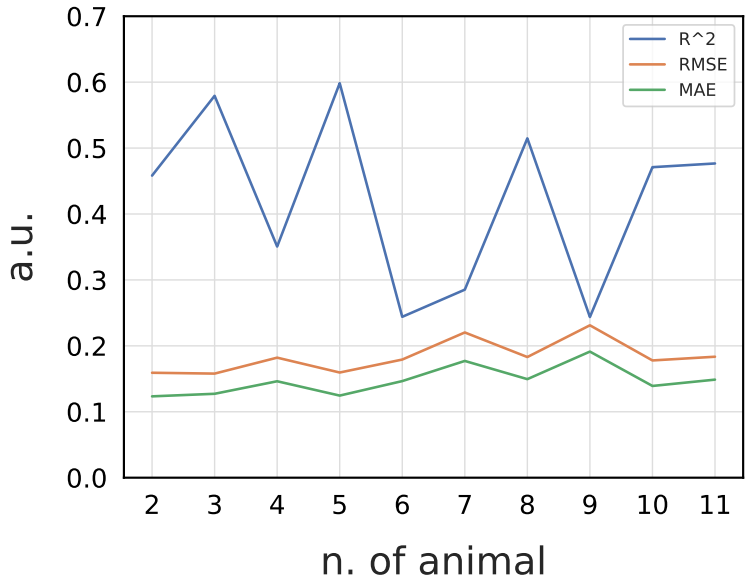
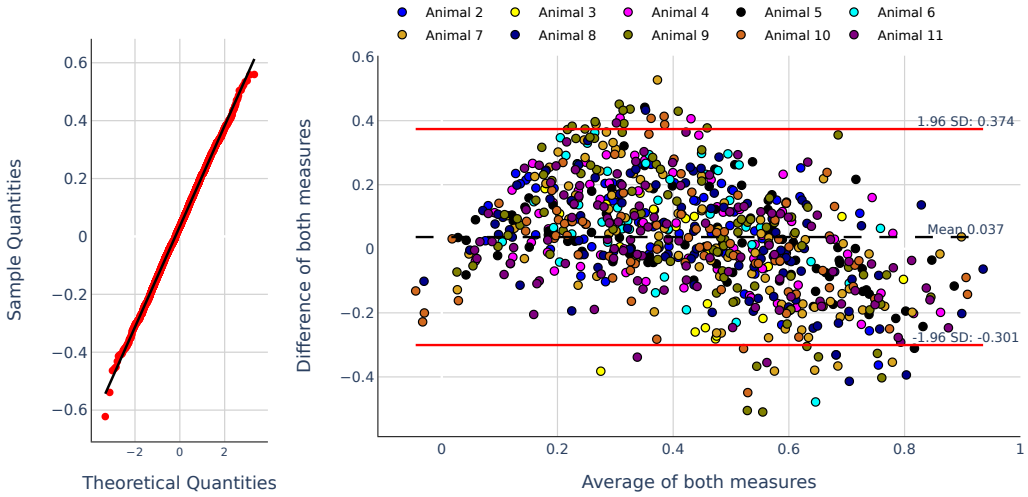



**Figure 4.6:** LOSOCV metrics for each animal of LinearGAM.

The graph highlights that animals 6, 7, and 9 display considerably lower $R^2$ values compared to others, leading to an overall lower $R^2$ metric with a high standard deviation. Although the standard deviation remains relatively consistent across other metrics, it is also observed to vary depending on the animal, as depicted in Figure 4.6. This issue may arise from the lack of calibration of individual animals during the data acquisition process, as each animal typically has unique characteristics. As $R^2$ decreased for certain animals, their stages were compared to determine whether they align with the stages of other animals. It was found that the stages are mostly similar in terms of the distribution of healthy and unhealthy stages, as well as the presence of blockages.

### 4.2.3 Regression Model Evaluation with Scatter Plots

The next type of model evaluation involves scatter plots, which helps in explaining assumptions of the statistical model, detecting biases, heteroskedasticity, or identifying outliers. The Bland-Altman plot, and the Q-Q plot (both displayed in Figure 4.7) and the residual histogram (shown in Figure 4.8) were used to assess the LinearGAM model and identify the relationships between the data points and the dependence of the animal.




**(a) :** Q-Q Plot **(b) :** Bland-Altman Plot

**Figure 4.7:** Evaluation plots: (a) represents the Q-Q plot, (b) represents the Bland-Altman plot, and each color represents one animal.

Figure 4.7a displays the quantile-quantile graph (Q-Q plot), which helps to assess the distributional assumption of the dataset through a comparison of data points. The theoretical expectation is that the data follow a normal distribution, depicted by the black line. The red points indicate how well the sampled data align with this theoretical distribution. Any deviations from the black line signify a deviation from the assumed normal distribution. Analysis of this plot indicates that most data points closely follow the expected normal distribution. Examining this graph is valuable as it provides insights into the spread and central tendency of the predicted data points. The confirmation of a normal distribution in this case is essential for a reliable assessment of the results of the statistical model.

Figure 4.7b shows the Bland-Altman plot, which shows the difference between two measures - prediction and reference - plotted against the average of both measurements. The black line represents the mean difference, with deviations from this line indicating the presence of bias. The two red lines above and below the mean line denote the limits of agreement (LoA), signifying the range within which the difference between the two measures is expected to lie with a certain level of confidence. A narrow LoA suggests minimal variability between the predictions and the reference. In this plot, most of the data points cluster around the mean line, with some outside the LoA range. This may show the presence of slight bias and variability in the model, although there is generally good agreement between the measures. Data points above the LoA may be caused by the above described difference in data behavior for each animal or the need for a more robust model, potentially impacting the interpretability of the results.

The residual plot shown in Figure 4.8 helps assess the suitability of the statistical regression models. These residuals, which represent the difference between the observed and predicted values, are plotted against the predicted value of the IBS flow. They are valuable for identifying

heteroskedasticity, which can violate the assumption of the relationship between predictor variables and the response variable, leading to reduced reliability of the model outcomes and erroneous conclusions. In this context, it is visible that the overall distribution of residual points in the plot follows a normal distribution. Points are randomly scattered around the horizontal axis, representing the fitted values, suggesting that the assumptions of independence of residuals, homoscedasticity, which means constant variance, and linearity are present.

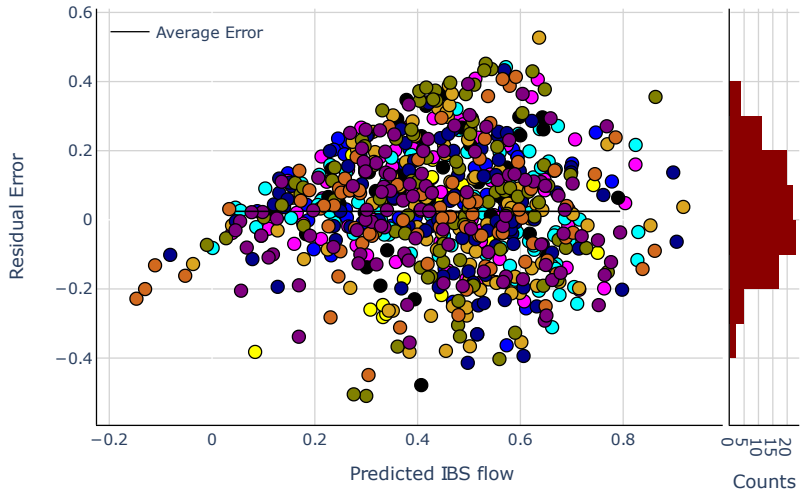



**Figure 4.8:** Residuals plot. Each colour represents one animal, the colour-coding is same as in Figure 4.7.

However, by showing the residual plot of each animal, some of them maintain a normal distribution, but the mean of the distribution is noticeably shifted from zero. Moreover, for animal 6, which had the worst $R^2$ in Figure 4.6, with the residual graph shown in Figure 4.9, its normal distribution slightly deviated. This indicates that while the overall residual plot satisfies all assumptions, there are certain animals that have heteroskedastic characteristics, which impact the overall model performance. This confirms the assumption highlighted in Figure 4.6, where numerical metrics and $R^2$ for each model were evaluated.

### 4.2.4 Results of Regression Models on Testing Animal

The final evaluation technique involves visually comparing the predictions on the testing animal. To provide a comprehensive comparison, the predictions of the baseline model are presented in Figure 4.10 in the middle of each row. Initially, a comparison was made between stages with **M** blockage (Figure 4.10b) and those without blockage (Figure 4.10a). In the healthy stage, the intensity of IBS flow in both lungs is weaker compared to the unhealthy stage. When comparing LinearGAM predictions, perfusion is more recognizable in the second case, when IBS flow is also higher. Differences can also be observed when comparing the results from the unhealthy stage of two measurements (Figures 4.10c and 4.10d). In Figure 4.10c, a distinct peak of perfusion in the left lung is clearly visible, while in Figure 4.10d, this peak is less evident, indicating a reduction in perfusion in the corresponding IBS flow signal, which could potentially cause problems. The perfusion of the lung is not recognizable in this figure.

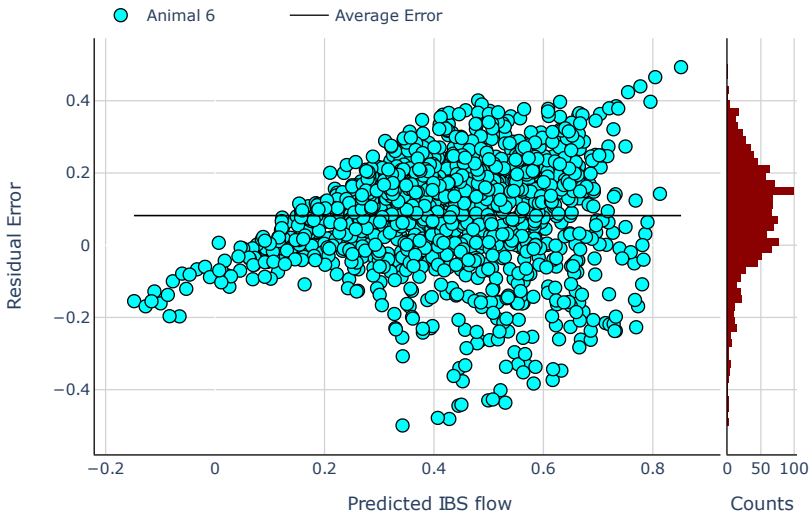


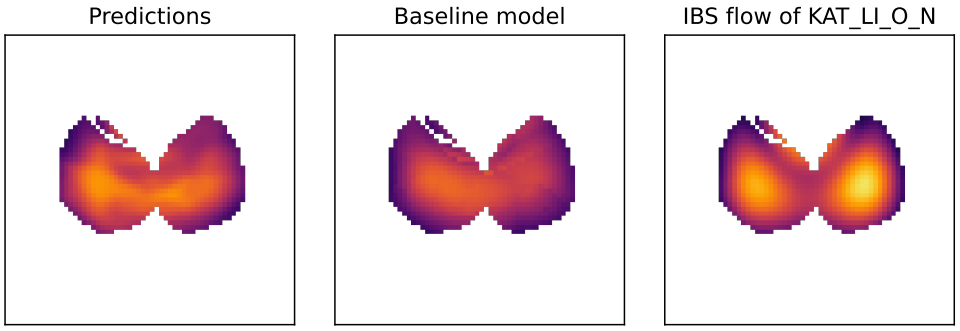

**Figure 4.9:** Residuals plot for animal 6. In previous graphs this animal has black colour, however for better visibility of individual data points the colour was changed.

Comparing the results of the LinearGAM predictions with the baseline model, which represents a state-of-the-art model, the LinearGAM model successfully identified lung perfusion in all cases except the one depicted in Figure 4.10d. Although the baseline model detected perfusion only in the first two described cases, it indicated a lower perfusion rate. From the visualization, it can be further observed that in the region between both lungs, there is an increase in amplitude 4.10a. This anomaly could be affected by the mixture of lung and heart EIT signal in this region. Consequently, due to the influence of the heart, the signal appears stronger, which leads the model to mistakenly predict a high rate of lung perfusion there.

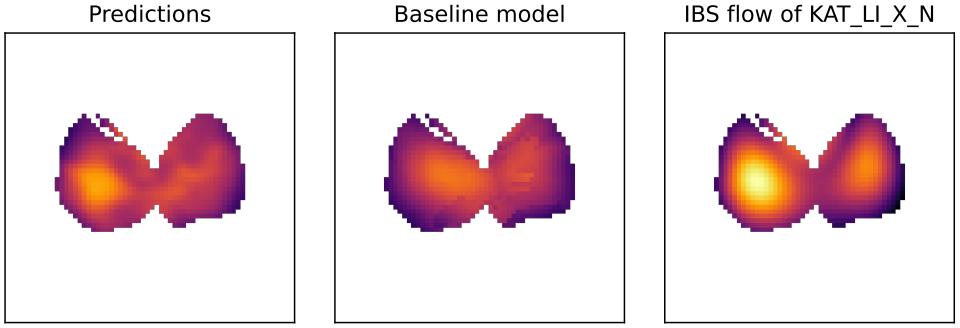
The evaluation of regression on testing animal concludes numerical results obtained from the testing animal. The evaluated models included LinearGAM, Random Forest, XGBoost, and a baseline model. All models underwent training on animals involved in the LOSOCV process and were subsequently tested on the testing animal to ensure unbiased results. The results are outlined in Table 4.5. The Random Forest model outperformed the others in all metrics ($R^2$, RMSE, and MAE), achieving an approximately 3% higher $R^2$ score compared to XGBoost. However, both models are considered non-interpretable. Specifically, the decision trees generated by Random Forest offer more interpretability than those produced by XGBoost. Furthermore, XGBoost performed around 4% better than LinearGAM, a model known for its high interpretability. It is noteworthy that RMSE and MAE are the same for both LinearGAM and XGBoost. In contrast, the baseline model lags behind all models, approximately 26% worse than LinearGAM in $R^2$, with significantly higher RMSE and MAE. These results underscore that the implemented models outperform the considered state-of-the-art model.

### 4.2.5 Regression Summary

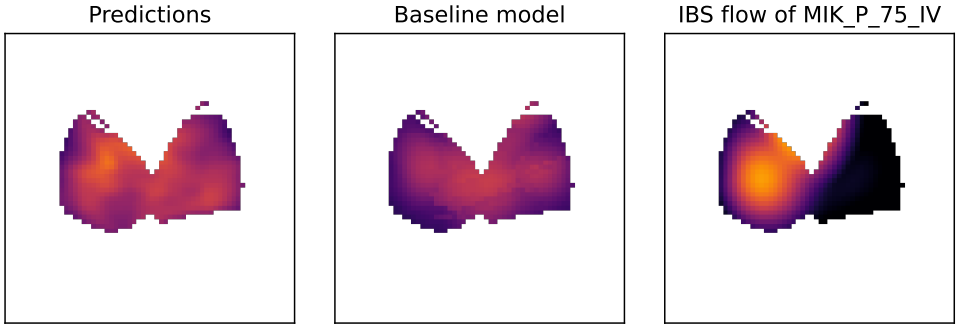
The dataset for the regression analysis contains both data points from lung and heart ROIs. Although the primary focus is on lung data, including perfusion measurements, heart data points are also necessary. During attempts to improve the results, it was found that excluding heart data leads to worse results. This is probably due to the fact that the dataset is not



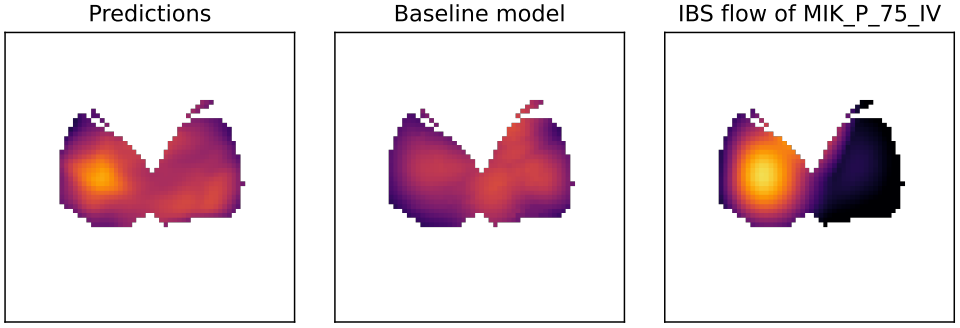

**(a) :** Healthy stage - no central blockage.




**(b) :** Unhealthy stage - central blockage.




**(c) :** Unhealthy stage - peripheral blockage, measurement 1.




**(d) :** Unhealthy stage - peripheral blockage, measurement 2.

**Figure 4.10:** Prediction visualization of regression results.

large enough after removing the heart data. To address this issue, heart data were included, but their impact was reduced by applying weights to prevent the model from focusing mainly on the peak amplitudes of these points. The weights of the heart ROIs are adjusted to half of the original value of the importance map, and the weights of the lung ROIs are 1.2 times higher. As depicted in Figure 4.11, this modification created the desired effect and shifted

| Model | $R^2$ | RMSE | MAE |
|---|---|---|---|
| LinearGAM | 0.56 | 0.17 | 0.14 |
| RandomForest | **0.63** | **0.15** | **0.12** |
| XGBoost | 0.60 | 0.17 | 0.14 |
| Baseline | 0.30 | 0.21 | 0.18 |

**Table 4.5:** Summary and evaluation metrics for regression models.

the peak values from the heart region to the center of each lung, helping the regression model to learn the perfusion patterns accurately and improve its performance.

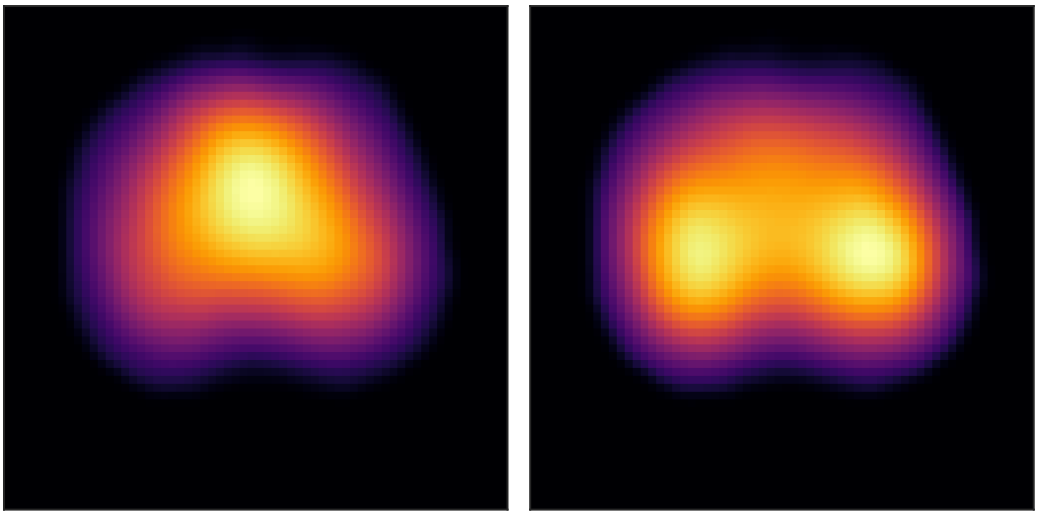


**Figure 4.11:** ROIs before and after using weights in regression. On the left side is the former importance map of the ROIs without using weights. On the right image, modified importance map used as weight map is depicted and lung ROIs are better recognizable than on the left side. Both images are from the same animal.

In conclusion, the regression evaluation metrics demonstrate the model's ability to recognize lung perfusion, as evidenced primarily in Figure 4.10. It is confirmed that less interpretable models result in better results, as observed in the numerical results outlined in Chapter 4.2.2. The scatter plots presented in Chapter 4.2.3 illustrate the preservation of linearity, homoscedasticity, and variability, crucial for valid and reliable results. However, variability in numerical results was observed among animals during LOSOCV, leading to poorer performance both numerically and visually in the final model. One potential solution to this issue could be calibrating each animal's measurement. Another improvement strategy could involve enhancing robustness by employing alternative methods that do not predict pixel-wise; for instance, methods that process the neighborhood of each pixel to derive its prediction.

## 4.3 Classification

This section presents an evaluation of the results of the second task of this thesis. Initially, the selected parameters for each model resulting from the Grid search algorithm are displayed. Subsequently, the numerical metrics, confusion matrices, and concluding visual representations are discussed for both binary and all forms of multi-class classification. Lastly, a summary of the classification results is provided.

### 4.3.1 Hyperparameter Value Selection

This section presents the best hyperparameter performance values obtained from the Grid Search algorithm. Table 4.6 displays the results for interpretable models, while Table 4.7 presents the results for non-interpretable models.

Using GAMs for multi-class classification is not feasible because the `pygam` library does not support it, as discussed in Chapter 3.3.6. Additionally, the implementation of weights is not possible in the classification case. Therefore, while GAM models are mentioned, they are not compared to other models.

| Model | DS | T | lambda | splines | n_iters | solver | penalty |
|---|---|---|---|---|---|---|---|
| LogisticGAM | No | B | 0.0215 | 20 | N/A | N/A | N/A |
| LogisticGAM | Yes | B | 0.0046 | 25 | N/A | N/A | N/A |
| LogReg | Yes/No | B/M | N/A | N/A | 1000 | lbfgs | l2 |

**Table 4.6:** The best parameters for interpretable models according to the Grid Search. The column name T stands for the type of classification, B means binary, M means multi-class.

| Model | DS | T | n_estimators | max_depth | learning_rate |
|---|---|---|---|---|---|
| RF | Yes/No | B/M | 1000 | 5 | N/A |
| XGBoost | Yes/No | B/M | 1000 | 3 | 0.1 |

**Table 4.7:** The best parameters for noninterpretable models according to the Grid Search. The column name T stands for the type of classification, B means binary, M means multi-class.

### 4.3.2 Binary Classification

Firstly, the results of binary classification are presented, focusing on classifying **H** and **U** data points. The numerical results and confusion matrices obtained from LOSOCV are initially presented, followed by an evaluation of the results obtained on testing data.

#### Numerical Results and Confusion Matrix

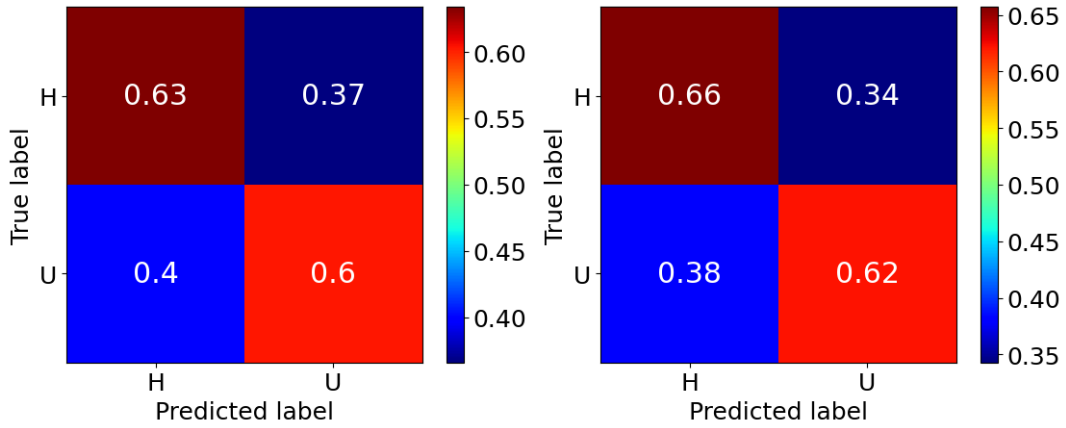
For binary classification, the results are summarized in Table 4.8. Each model was tested with no correction, downsampling, or weighting. Only LogisticGAM is listed for downsampling, as the results were very poor for other cases. The best models from Table 4.8 are further compared using the confusion matrices shown in Figure 4.12.

For each metric in Table 4.8, the two best models are highlighted. The primary metric considered is accuracy; however, it is crucial to note that accuracy may not be reliable in evaluating an imbalanced dataset, as consistently predicting the majority class can inflate accuracy. Therefore, other more suitable metrics are presented here, with the final assessment based on the F-1 score. The F-1 score combines precision and recall into a single value and is more reliable in handling unbalanced datasets. It is evident that models with higher accuracy tend to have poorer F-1 scores, especially those that did not utilize downsampling or weighting techniques, indicating class imbalance in the dataset. The two models with the highest F-1 scores, both with downsampled datasets, are logistic regression (61.29%) and random forest classifier (62.55%). The third-best model is LogisticGAM, which also showed satisfactory performance with an F-1 score of 59.31%.

The confusion matrices for both methods are depicted in Figure 4.12. Although the RF model exhibits better results, it is non-interpretable. Since the results of the logistic regression

| Model | DS | W | Accuracy[%] | Precision[%] | Recall[%] | F-1 score [%] |
|---|---|---|---|---|---|---|
| LogisticGAM | Yes | No | 60.63 | 63.01 | 56.01 | 59.31 |
| LogReg | No | No | **81.93** | 20.8 | 1.22 | 2.29 |
| LogReg | No | Yes | 58.87 | 24.59 | 58.04 | 34.55 |
| **LogReg** | Yes | No | 62.14 | 62.26 | **60.35** | **61.29** |
| RF | No | No | 70.74 | 31.50 | 47.21 | 37.78 |
| RF | No | Yes | 65.87 | 29.45 | 56.40 | 38.69 |
| **RF** | Yes | No | 63.98 | **66.2** | **59.29** | **62.55** |
| XGB | No | No | **80.51** | 40.32 | 20.64 | 27.30 |
| XGB | No | Yes | 72.73 | 32.83 | 46.40 | 38.45 |
| XGB | Yes | No | 62.63 | **67.63** | 48.39 | 56.42 |
| Baseline | Yes | No | 59.08 | 52.56 | 62.95 | 54.61 |

**Table 4.8:** Model evaluation metrics for binary classification. DS stands for downsampling, W stands for weighting. The metrics represent successfully indicated **U** class.




**(a) :** LogReg CM **(b) :** RF CM

**Figure 4.12:** Confusion matrices for binary classification. (a) represents the confusion matrix for the logistic regression model, (b) represents the matrix for the RF model.

are very similar, they are utilized in the visualizations in Chapter 4.3.2. Comparing the F-1 score of both most successful models with the baseline model, which is considered the state-of-the-art model, LogReg performed better by 7%, and RF performed better by 8%. This low F-1 score could be attributed to the low variability in amplitude between **H** and **U**, removal of important data points by downsampling, or the confusion of the model with data points at the lung edges in the **H** class with low amplitude, similar to the **U** class. However, downsampling proved to be a more effective technique than weighting in cases of substantial class imbalance.

### Results of Classification Models on Testing Animal

From Figure 4.13, it is evident that, while there are slight variations in the numerical results evaluated using LOSOCV, these differences are more noticeable in the prediction visualization. When comparing healthy cases in 4.13a and 4.13c, Random Forest demonstrates a more precise classification compared to Logistic Regression. LogReg tends to label more data points as unhealthy. In both predictions, edge data points marked as unhealthy are visible, even if they should be predicted as healthy. This problem is likely due to the decreased amplitude

of IBS flow in the peripheries of the lung, which causes models to misclassify them. For the unhealthy cases, RF correctly identifies the peripheral blockage almost entirely, and in healthy lungs, it only marks a small area at the edges as unhealthy. However, LogReg struggles to identify blockages throughout the lung and also misclassifies more unhealthy data points in healthy lungs. When comparing both models with the baseline model, both of them still have better visualization results on the test animal.

Analysis of Table 4.9 reveals that the better outcomes were achieved by the less interpretable models. The better of them, XGBoost, outperformed the LogReg model by approximately 4% in F-score and the baseline model by approximately 8%. By comparing the LogReg model with the baseline model, the LogReg performs better about 4%.

### 4.3.3 4-class Classification

This section outlines the approach used to classify data points into four distinct classes: **H**, **C**, **M**, and **P**. Initially, numerical results and confusion matrices obtained from the LOSOCV method are presented. Subsequently, the models are evaluated on a testing animal, and the outcomes of the methods are compared.

| Model | Accuracy | Precision | Recall | F-1 score |
|---|---|---|---|---|
| LogReg | 46.43 | 25.11 | 66.67 | 36.48 |
| RandomForest | 57.01 | 28.40 | 56.76 | 37.86 |
| XGBoost | 70.21 | 37.65 | 44.36 | 40.73 |
| Baseline | 45.23 | 22.79 | 57.52 | 32.64 |

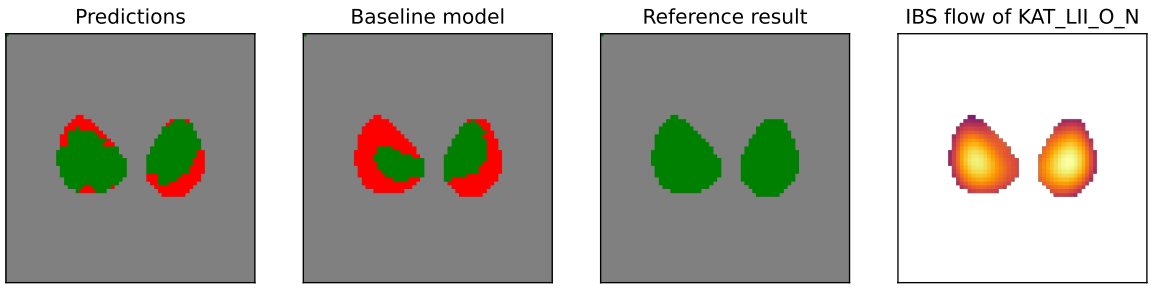
**Table 4.9:** Model evaluation metrics for binary classification on testing animal.

#### Numerical Results and Confusion Matrix

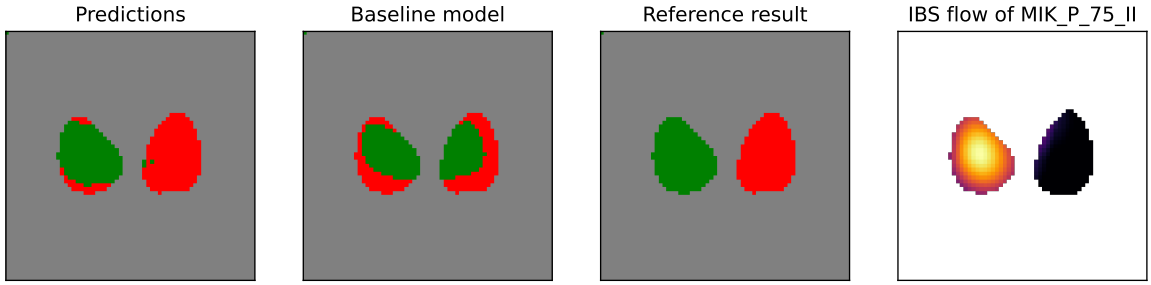
In the multi-class classification, the downsampling technique has been used to enhance model performance due to unbalanced datasets. This approach has proven to be effective not only in binary classification but also in multi-class scenarios. Additionally, weighting has been utilized to address the removal of many data points from the dataset, particularly those belonging to minority classes. The results of the multiclass classification are described in Table 4.11, with the F-1 score being an important metric for evaluating the 4-class classification due to the imbalance in the dataset. While XGBoost is dominant for two classes, Random Forest was chosen as the selected non-interpretable model due to its consistent performance across all classes. Among the interpretable models, logistic regression outperformed the baseline model, which replaces a non-existent state-of-the-art.

When comparing RF and LogReg, random forest achieved approximately a 5% higher F-1 score for healthy class and a 13% higher score for the peripheral class. However, both models achieved similar results for the central and medial classes. Although the baseline model's highest F-1 score is for the **H** class, LogReg outperformed it by approximately 7%, while RF outperformed it by approximately 20%.

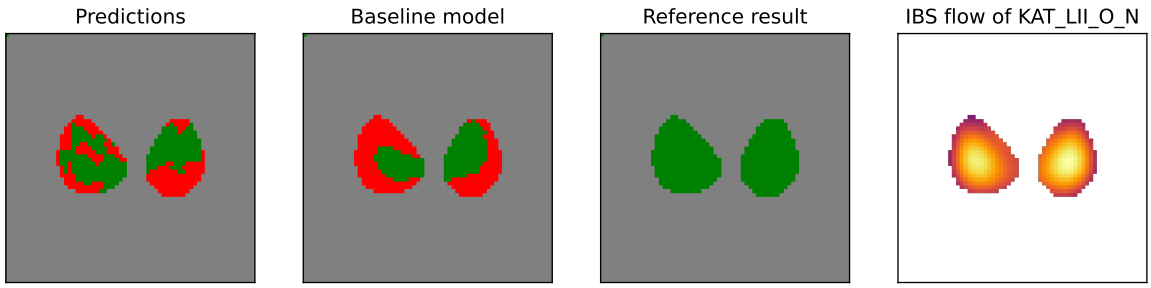
The confusion matrices shown in Figure 4.14 provide more detailed insight into the prediction results of both the LogReg and RF methods. These matrices confirm the previous observation of similar outcomes for the **C** and **M** classes, as detailed in Table 4.11. Furthermore, it is evident that the LogReg method performs better for the **C** and **M** classes, as demonstrated by the confusion matrix in Figure 4.14a, compared to the RF method depicted in Figure 4.14b. In contrast, the RF method has significantly better results on the healthy and peripheral classes.



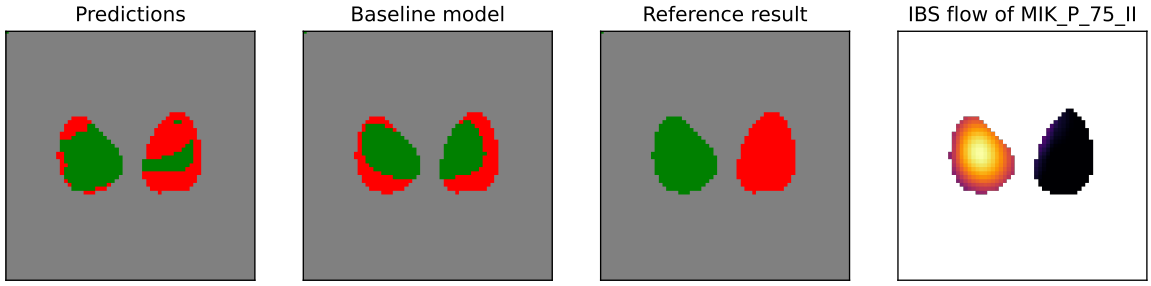

**(a) :** Random Forest - healthy.




**(b) :** Random Forest - peripheral blockage.




**(c) :** Logistic Regression - healthy.




**(d) :** Logistic regression - peripheral blockage.

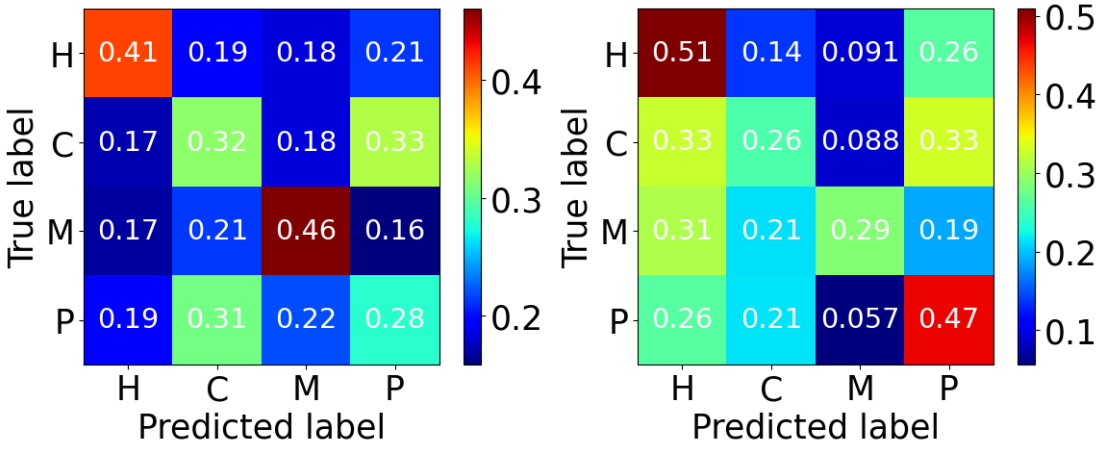
**Figure 4.13:** Visualization of binary classification prediction. The red color represents the unhealthy part, and the green color represents the healthy part of the lungs.

### Results of Classification Models on Testing Animal

In this section, the final visualization of model predictions for multi-class classification is performed on the test animal, depicted in Figure 4.15. This comparison focuses on the RF and LogReg methods. In the visualization of IBS flow shown in Figures 4.15a and 4.15b, the left lung exhibits a lower amplitude. When analyzing the prediction results from LogReg, areas with lower amplitudes are predominantly classified as **C** blockages, whereas RF has classified the areas with decreased amplitude correctly. This inaccuracy results from labels,

| Class | Model | Accuracy[%] | Precision[%] | Recall[%] | F-1 Score[%] |
|---|---|---|---|---|---|
| Healthy | LogReg | 38.90 | 69.06 | 41.16 | 51.56 |
| | RF | 44.94 | 63.96 | 50.94 | **56.71** |
| | XGB | 46.86 | 56.44 | 72.66 | **63.53** |
| | Baseline | 26.50 | 58.13 | 36.10 | 44.54 |
| Central | LogReg | 38.90 | 17.59 | 31.62 | **22.60** |
| | RF | 44.94 | 19.16 | 25.67 | **21.94** |
| | XGB | 46.86 | 9.44 | 5.37 | 6.85 |
| | Baseline | 26.50 | 1.89 | 0.5 | 0.8 |
| Medial | LogReg | 38.90 | 16.44 | 46.05 | **24.23** |
| | RF | 44.94 | 23.06 | 28.76 | **25.57** |
| | XGB | 46.86 | 8.13 | 5.79 | 6.77 |
| | Baseline | 26.50 | 10.04 | 57.60 | 17.10 |
| Peripheral | LogReg | 38.90 | 32.98 | 27.91 | 30.23 |
| | RF | 44.94 | 41.36 | 46.63 | **43.85** |
| | XGB | 46.86 | 43.15 | 33.68 | **37.83** |
| | Baseline | 26.50 | 12.22 | 9.06 | 10.40 |

**Table 4.10:** Model evaluation metrics for 4-class classification.



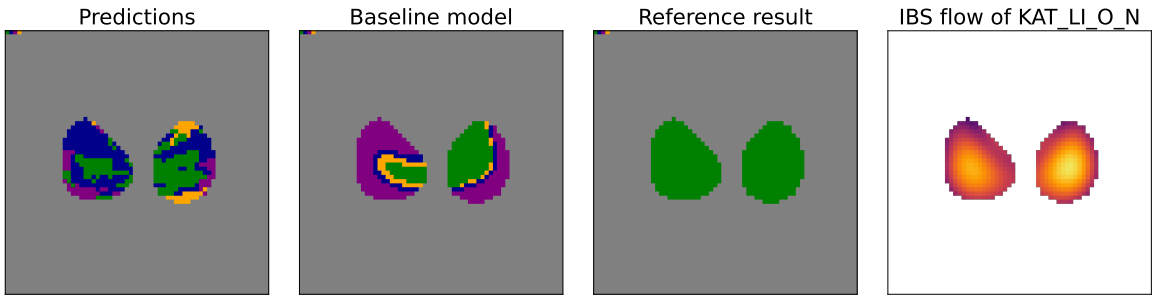

**(a) :** LogReg CM **(b) :** RF CM

**Figure 4.14:** Confusion matrices for 4-class classification. (a) represents the confusion matrix for the logistic regression model and (b) represents the matrix for the RF model.

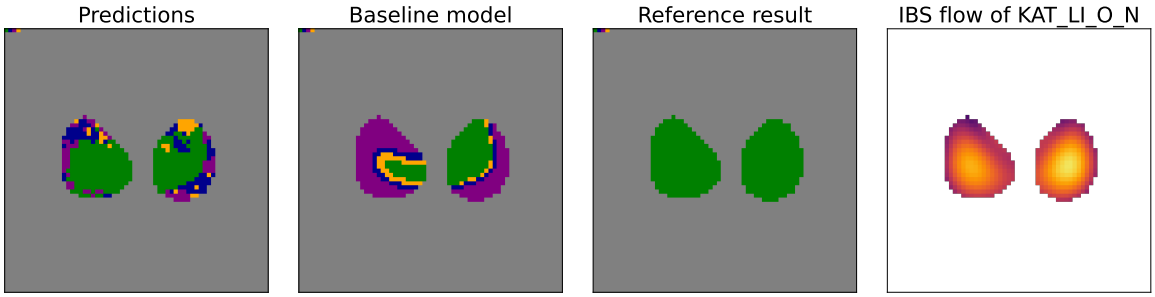
where such low amplitude values are often one type of unhealthy, possibly indicating **C** or **M** blockages, which LogReg struggles to address. However, LogReg correctly identifies the center of the lungs, typically associated with **C** or **M** blockages, as healthy. The problems with incorrectly classified lung edges were discussed in Chapter 4.3.2.

Another notable observation is that both methods sometimes misclassify **C** blockages as other types. LogReg, for instance, classifies **M** blockage as **C**, as shown in Figure 4.15c, whereas RF categorizes the same blockages as **P**, as demonstrated in Figure 4.15d. Overall, the random forest method achieves better visualization results. When comparing both methods with the baseline model, it is evident that they achieve higher precision in classifying data points.

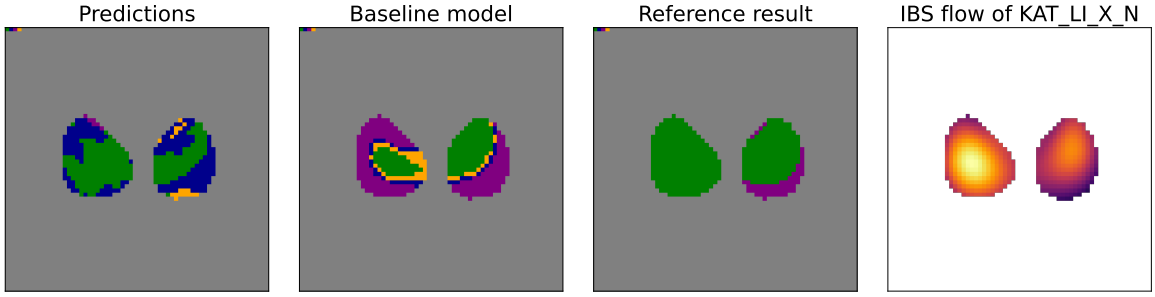
Finally, numerical metrics for the testing animal were obtained and are presented in Figure 4.11. In most cases, XGBoost outperforms other models, but shows the lowest F-1



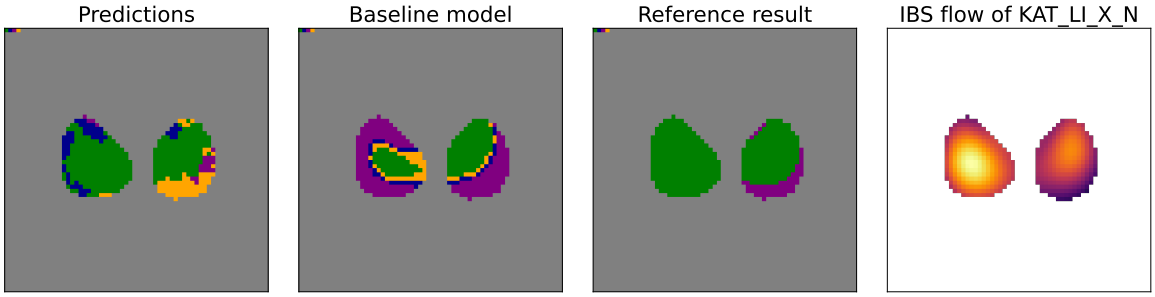

**(a) :** Logistic Regression - healthy.




**(b) :** Random Forest - healthy.




**(c) :** Logistic Regression - medial blockage.




**(d) :** Random Forest - medial blockage.

**Figure 4.15:** Visualization of 4-class classification prediction. Blue represents the central blockage, purple represents the medial blockage, and yellow represents the peripheral blockage of the lungs.

score in the medial class, where the baseline model achieved the best score among all considered models. In general, the best results were observed for all methods in classifying healthy and peripheral classes. The most significant discrepancy in model performances is seen in the peripheral class, where the baseline model has a roughly 35% lower F-1 score than the others.

Due to the similarity and anatomical proximity of the central and peripheral classes, they were merged into one class, aiming to enhance the models' performance in classifying these blockages.

| Class | Model | Accuracy[%] | Precision[%] | Recall[%] | F-1 Score[%] |
|---|---|---|---|---|---|
| Healthy | LogReg | 45.80 | 87.04 | 45.11 | **59.42** |
| | RF | 41.17 | 86.73 | 39.37 | 54.15 |
| | XGB | 73.35 | 82.21 | 87.53 | **84.78** |
| | Baseline | 34.41 | 75.85 | 39.19 | 51.68 |
| Central | LogReg | 45.80 | 14.53 | 60.16 | **23.40** |
| | RF | 41.17 | 12.46 | 53.92 | 20.24 |
| | XGB | 73.35 | 28.37 | 21.25 | **24.30** |
| | Baseline | 34.41 | 5.81 | 6.64 | 6.20 |
| Medial | LogReg | 45.80 | 5.13 | 7.88 | **6.21** |
| | RF | 41.17 | 4.22 | 6.23 | 5.30 |
| | XGB | 73.35 | 0.79 | 0.18 | 0.30 |
| | Baseline | 34.41 | 7.33 | 55.50 | **12.95** |
| Peripheral | LogReg | 45.80 | 30.89 | 61.78 | **41.12** |
| | RF | 41.17 | 27.34 | 56.50 | 38.57 |
| | XGB | 73.35 | 39.74 | 45.02 | **42.21** |
| | Baseline | 34.41 | 7.11 | 6.48 | 6.78 |

**Table 4.11:** Model evaluation metrics for 4-class classification on testing animal.

### 4.3.4 3-class classification

After identifying a classification challenge between central and medial blockages, these two categories were merged into one. This section discusses the results of the 3-class classification.

#### Numerical Results and Confusion Matrix

Table 4.12 presents the numerical results obtained using the LOSOCV method for an unbiased comparison. Similarly to 4-class classification, the random forest method demonstrates the most stable results across all classes. Although the results between random forest and logistic regression differ by about 5%. These differences are the smallest among all previous classification tasks between and interpretable and non-interpretable models. The baseline model achieves the best results in the merged class but performs poorly in the peripheral class. It can be concluded that logistic regression remains the most stable method as an interpretable model, while random forest stands out as a non-interpretable model.

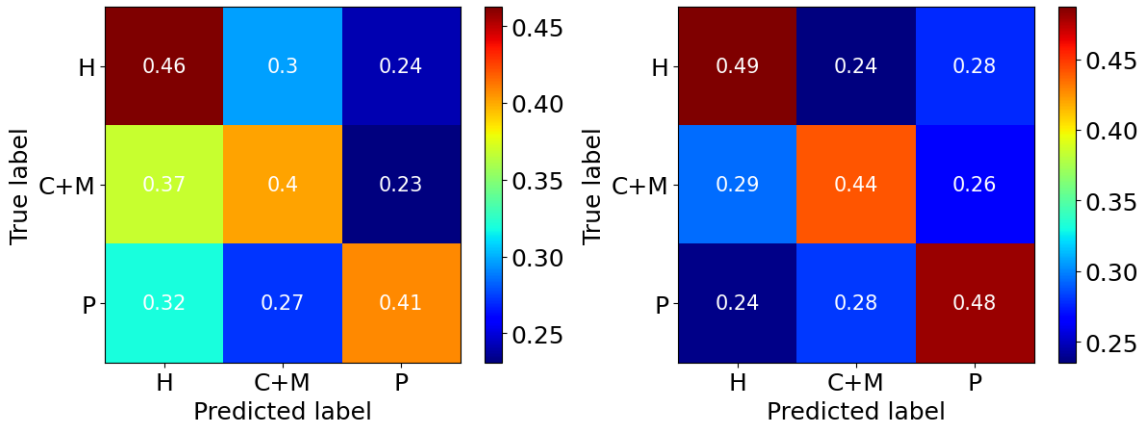
The assumptions are also proved by Figure 4.16. In the confusion matrix of LogReg shown in Figure 4.16a, the recall on the diagonal confirms the lower ability of logistic regression to classify **C+M** and **P** classes as they are a bit suppressed by **H**. However, the confusion matrix shown in Figure 4.16a shows that merging the central and peripheral classes helps the random forest method to obtain balanced results in all classes.

#### Results of Classification Models on Testing Animal

This subsection presents the evaluation results on the testing animal. First, the results of RF and LogReg are depicted in Figure 4.17 to compare the classification results on the same data. RF, as shown in Figure 4.17b, classifies the data points mainly on the edges of the lung, where IBS flow decreases. On the contrary, LogReg categorizes more data points at the bottom of both lungs as having **C** or **M** blockage, while edges are less misclassified than for RF. As illustrated in these two figures, each method has a different prediction approach. Therefore, finding one method capable of addressing all of the issues remains a challenge,

| Class | Model | Accuracy[%] | Precision[%] | Recall[%] | F-1 Score[%] |
|---|---|---|---|---|---|
| Healthy | LogReg | 46.33 | 40.22 | 46.26 | 43.03 |
| | RF | 50.78 | 47.78 | 48.71 | **48.24** |
| | XGB | 37.14 | 37.81 | 78.38 | **51.01** |
| | Baseline | 33.79 | 41.76 | 44.28 | 42.98 |
| Central + Medial | LogReg | 46.33 | 41.35 | 40.11 | 40.72 |
| | RF | 50.78 | 46.09 | 44.14 | **45.09** |
| | XGB | 37.14 | 35.13 | 13.89 | 19.91 |
| | Baseline | 33.79 | 38.57 | 59.46 | **46.79** |
| Peripheral | LogReg | 46.33 | 46.37 | 40.81 | **43.41** |
| | RF | 50.78 | 47.02 | 48.09 | **47.55** |
| | XGB | 37.14 | 48.66 | 25.85 | 33.77 |
| | Baseline | 33.79 | 7.05 | 2.80 | 4.01 |

**Table 4.12:** Model evaluation metrics for 3-class classification.




**(a) :** LogReg CM

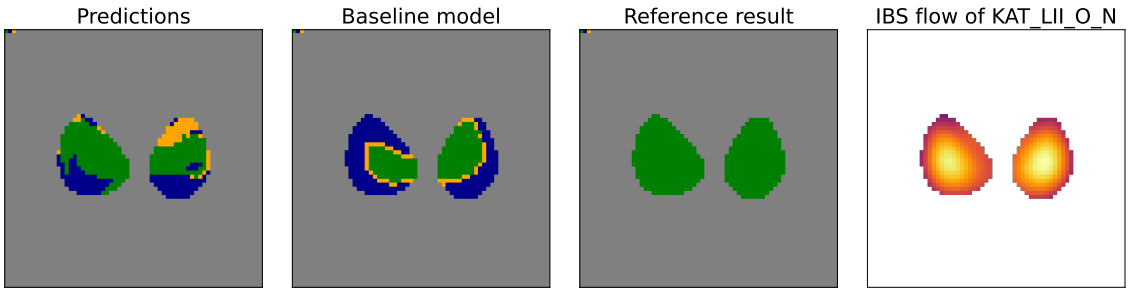
**(b) :** RF CM

**Figure 4.16:** Confusion matrices for 3-class classification. (a) represents the confusion matrix for the logistic regression model, (b) represents the matrix for the RF model.

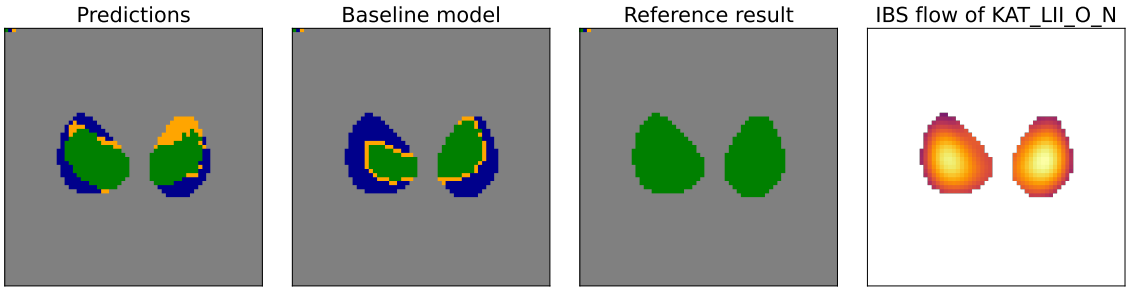
especially in the cases of unbalanced datasets with known differences in the number of data points for each class.

The other interesting outcomes are depicted in Figure 4.18. In this comparison, two measurements from one stage are evaluated. In the reference result column, it is evident that the IBS flow amplitude in the second measurement is slightly higher (highlighted in green) than the threshold for recognizing the data points as unhealthy. Initially, in the output in Figure 4.18ait was correctly classified as **C+M** blockage, although the defined area is considerably smaller than the reference area. On the contrary, in the results shown in Figure 4.18b, the classified blockage area is larger, but the model misclassified it as a peripheral blockage, which is incorrect. This comparison highlights that, despite two measurements of the same animal and stage, the reference IBS flow differs a lot as well as the EIT measurements. The variance of the EIT measurements and respective extracted features can cause such misclassifications. Hence, calibration may be necessary for each measurement to obtain the most accurate data. In this particular case, the baseline model outperformed logistic regression. However, in general, logistic regression achieves better results than the baseline model.

Furthermore, as in the previous classification types, the 3-class classification was assessed

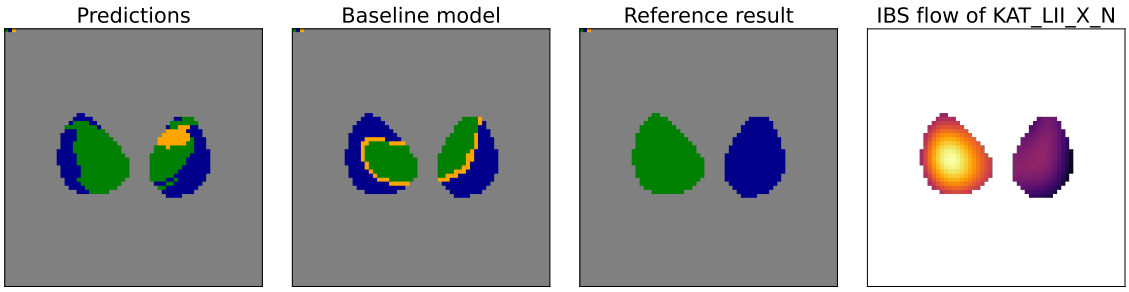



**(a) :** Logistic Regression - healthy.

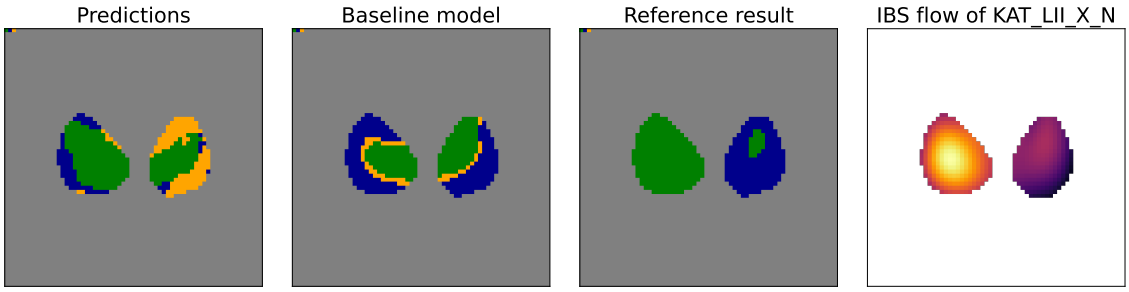



**(b) :** Random Forest - healthy.

**Figure 4.17:** Visualization of 3-class classification prediction. Blue represents the central + medial blockage and yellow represents the peripheral blockage of the lungs.




**(a) :** Logistic regression - C+M blockage, measurement 1.




**(b) :** Logistic regression - C+M blockage, measurement 2.

**Figure 4.18:** Visualization of 3-class classification prediction, comparison of two measurements of the same stages. Blue represents the central + medial blockage, and yellow represents the peripheral blockage of the lungs.

using evaluation techniques. Random Forest and logistic regression achieved the most favorable results on unhealthy classes. For the testing animal, the baseline model also performed well, except for the peripheral class, where its F-1 score was only 8%.

| Class | Model | Accuracy[%] | Precision[%] | Recall[%] | F-1 Score[%] |
|---|---|---|---|---|---|
| Healthy | LogReg | 54.29 | 89.91 | 52.01 | 65.60 |
| | RF | 54.91 | 90.23 | 54.61 | **68.04** |
| | XGB | 65.72 | 81.24 | 77.49 | **79.32** |
| | Baseline | 40.11 | 75.82 | 38.95 | 51.46 |
| Central + Medial | LogReg | 54.29 | 20.47 | 57.97 | **30.26** |
| | RF | 54.91 | 23.81 | 41.65 | **30.29** |
| | XGB | 65.72 | 23.53 | 26.10 | 24.74 |
| | Baseline | 40.11 | 19.07 | 69.44 | 29.93 |
| Peripheral | LogReg | 54.29 | 37.03 | 67.30 | **47.78** |
| | RF | 54.91 | 24.41 | 75.58 | **36.90** |
| | XGB | 65.72 | 22.27 | 27.07 | 24.44 |
| | Baseline | 40.11 | 7.34 | 8.92 | 8.05 |

**Table 4.13:** Model evaluation metrics for 3-class classification on testing animal.

### 4.3.5 Classification Summary

In the classification section, various approaches were introduced and assessed. Initially, binary classification was employed to assess label distributions and determine a strategy for multi-class classification. Subsequently, a 4-class classification was performed to differentiate between all unhealthy classes. As the amount of **H** class was much higher, it was downsampled to match the total of the three unhealthy labels (**C**, **M**, **P**), while the remaining labels were adjusted by weighting to achieve a balanced dataset. However, this adjustment was not effective for the 4-class scenario, due to the significantly lower occurrence of the **C** and **M** labels compared to the other classes. Consequently, these two classes were merged into a single class, as they are anatomically close and produce similar signal characteristics. Although this method proved to be the most effective approach, it is important to note that achieving this required the removal of a large number of data points.

# Chapter 5

# Conclusion and Outlook

Electrical impedance tomography (EIT) offers valuable insights into regional lung perfusion, a crucial aspect of respiratory function. The objective of this thesis is to develop a real-time and non-invasive method for assessing lung perfusion using the cardiac component of the EIT recordings (CRS). Currently, the state-of-the-art method for perfusion monitoring involves utilizing the contrast signal (IBS) obtained through intravenous saline injection. However, this method is invasive, potentially harmful, and requires offline analysis. In contrast, our proposed method focuses on leveraging the CRS signal along with the development of statistical models and selected signal features with physiological significance.

To address this objective, two statistical tasks were employed: regression to quantify lung perfusion levels and classification to assess the type of lung blockage. Statistical models were trained using data from animals with various perfusion insufficiencies, using gold standard IBS flow as reference. The data were collected from 11 animals, each animal with several stages of health and various types of lung injuries. One animal was exclude from training for unbiased model evaluation, while the remaining animals were trained, validated, and tested using leave-one-subject-out cross-validation (LOSOCV). To correctly assess the performance of the regression and classification tasks, evaluation metrics for both were used.

Due to the application of statistical models to medical data, one of the requirements was high degree of model interpretability. Therefore, in the regression task, the LinearGAM model was employed for this purpose. For classification, GAM models were used for binary classification and Logistic regression for multi-class classification. However, for a comprehensive comparison of results, non-interpretable models such as Random Forests and XGBoost were also implemented for all tasks. Moreover, since there is no existing state-of-the-art model for any of the previously mentioned tasks, a baseline model was established as a replacement. It is a linear model trained on the most significant features with respect to IBS flow, healthy / unhealthy state, or type of blockage, depending on the task.

In the regression task, both interpretable and non-interpretable models demonstrated the ability to recognize lung perfusion, as evidenced by metrics obtained through LOSOCV and validation on testing animal. As expected, the less interpretable models outperformed the interpretable ones, although the difference was not significant. Specifically, LinearGAM achieved the $R^2$ score of 0.43, while XGBoost achieved 0.47. However, in regression, the difference between the baseline model and LinearGAM was substantial. Comparing the LinearGAM with the baseline model, the baseline model was outperformed in $R^2$ by 0.41, while the best non-interpretable model, XGBoost, was around 0.45 in $R^2$ score better than the baseline model.

Subsequently, the classification problem was solved. In binary classification, models sufficiently correct distinguished between healthy and unhealthy data points. Following this, a multi-class classification approach was implemented. Initially, a 4-class classification was

used to classify healthy data and three types of blockages: central, medial, and peripheral. However, due to the limited number of data points for central and medial blockages, a 3-class classification that merges these two classes was introduced. Evaluation and comparison of the models were based on the F-1 score. The Random Forest model achieved the most stable results, with an average F-1 score of 46% for all classes, while Logistic Regression, as an interpretable model, also had a comparable performance of 43%. The baseline model performed nearly as well as Logistic Regression in classifying the healthy class, but struggled in classifying blockages.

Since the results of both tasks are very promising, further improvements are possible. During the implementation of statistical models and the evaluation of their results, variance in model performance was observed. The evaluation metrics of each model in LOSOCV indicated that some animals gave worse results compared to others. Given that the performance pattern for all animals in each task was consistent, calibration of the measurements of each animal might be necessary. As some classes are represented by more data points than others, a more balanced dataset could potentially help models achieve better results without the need of downsampling, which removes large amounts of data points. Lastly, a more robust approach could be employed, such as considering a pixel with its neighborhood as one data point.

In conclusion, this thesis effectively utilized CRS signals for the identification and characterization of lung perfusion and injuries, while also highlighting the limitations of this approach. Using these results and conducting additional investigations, it has the potential to develop a non-invasive and real-time technique to provide information on lung perfusion.

# Bibliography


[1] C. Putensen, B. Hentze, S. Muenster, and T. Muders, "Electrical Impedance Tomography for Cardio-Pulmonary Monitoring," *Journal of Clinical Medicine*, vol. 8, p. 1176, Aug. 2019.

[2] B. Brown, "Electrical impedance tomography (EIT): a review," *Journal of Medical Engineering & Technology*, vol. 27, pp. 97–108, Jan. 2003.

[3] B. Hentze, T. Muders, H. Luepschen, E. Maripuu, G. Hedenstierna, C. Putensen, M. Walter, and S. Leonhardt, "Regional lung ventilation and perfusion by electrical impedance tomography compared to single-photon emission computed tomography," *Physiological Measurement*, vol. 39, p. 065004, June 2018.

[4] A. Vonk Noordegraaf, P. W. A. Kunst, A. Janse, J. T. Marcus, P. E. Postmus, T. J. C. Faes, and P. M. J. M. De Vries, "Pulmonary perfusion measured by means of electrical impedance tomography," *Physiological Measurement*, vol. 19, pp. 263–273, May 1998.

[5] J. B. Borges, F. Suarez-Sipmann, S. H. Bohm, G. Tusman, A. Melo, E. Maripuu, M. Sandström, M. Park, E. L. V. Costa, G. Hedenstierna, and M. Amato, "Regional lung perfusion estimated by electrical impedance tomography in a piglet model of lung collapse," *Journal of Applied Physiology*, vol. 112, pp. 225–236, Jan. 2012.

[6] A. Adler and D. Holder, *Electrical Impedance Tomography: Methods, History and Applications*. Boca Raton: CRC Press, 2 ed., Nov. 2021.

[7] S. Leonhardt and B. Lachmann, "Electrical impedance tomography: the holy grail of ventilation and perfusion monitoring?," *Intensive Care Medicine*, vol. 38, pp. 1917–1929, Dec. 2012.

[8] S. Khalil, M. Mohktar, and F. Ibrahim, "The Theory and Fundamentals of Bioimpedance Analysis in Clinical Status Monitoring and Diagnosis of Diseases," *Sensors*, vol. 14, pp. 10895–10928, June 2014.

[9] C. J. Kotre, "EIT image reconstruction using sensitivity weighted filtered backprojection," *Physiological Measurement*, vol. 15, pp. A125–A136, May 1994.

[10] R. S. Tavares, F. A. N. Filho, M. S. Tsuzuki, T. C. Martins, and R. G. Lima, "Discretization Error and the EIT Forward Problem," *IFAC Proceedings Volumes*, vol. 47, no. 3, pp. 7535–7540, 2014.

[11] A. Adler, J. H. Arnold, R. Bayford, A. Borsic, B. Brown, P. Dixon, T. J. C. Faes, I. Frerichs, H. Gagnon, Y. Gärber, B. Grychtol, G. Hahn, W. R. B. Lionheart, A. Malik,

R. P. Patterson, J. Stocks, A. Tizzard, N. Weiler, and G. K. Wolf, "GREIT: a unified approach to 2D linear EIT reconstruction of lung images," *Physiological Measurement*, vol. 30, p. S35, June 2009.

[12] G. Hahn, I. Sipinkova, F. Baisch, and G. Hellige, "Changes in the thoracic impedance distribution under different ventilatory conditions," *Physiological Measurement*, vol. 16, pp. A161–A173, Aug. 1995.

[13] K. A. Powers and A. S. Dhamoon, "Physiology, Pulmonary Ventilation and Perfusion," in *StatPearls*, Treasure Island (FL): StatPearls Publishing, 2024.

[14] B. Hentze, T. Muders, H. Luepschen, S. Leonhardt, C. Putensen, and M. Walter, "Gamma-variate modeling of indicator dilution curves in electrical impedance tomography," in *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, (Seogwipo), pp. 3596–3599, IEEE, July 2017.

[15] H. K. Thompson, C. F. Starmer, R. E. Whalen, and H. D. Mcintosh, "Indicator Transit Time Considered as a Gamma Variate," *Circulation Research*, vol. 14, pp. 502–515, June 1964.

[16] B. H. Brown, A. Leathard, A. Sinton, F. J. McArdle, R. W. M. Smith, and D. C. Barber, "Blood flow imaging using electrical impedance tomography," *Clinical Physics and Physiological Measurement*, vol. 13, pp. 175–179, Dec. 1992.

[17] C. A. Grant, T. Pham, J. Hough, T. Riedel, C. Stocker, and A. Schibler, "Measurement of ventilation and cardiac related impedance changes with electrical impedance tomography," *Critical Care*, vol. 15, p. R37, Jan. 2011.

[18] A. Adler, M. Proença, F. Braun, J. X. Brunner, and J. Solà, "Origins of Cardiosynchronous Signals in EIT,"

[19] S. Boslaugh and P. A. Watters, *Statistics in a nutshell*. Farnham: O'Reilly, 2008.

[20] S. Turney, "Pearson correlation coefficient (r) | guide examples." `https://www.scribbr.com/statistics/pearson-correlation-coefficient/`, Jan. 2024.

[21] "Spearman's rank-order correlation." `https://statistics.laerd.com/statistical-guides/spearmans-rank-order-correlation-statistical-guide.php`, Jan. 2024.

[22] V. Nasteski, "An overview of the supervised machine learning methods," *HORIZONS.B*, vol. 4, pp. 51–62, Dec. 2017.

[23] "supervised learning python." `https://pythonspot.com/supervised-learning/`, Dec. 2023.

[24] W. Loh, "Classification and regression trees," *WIREs Data Mining and Knowledge Discovery*, vol. 1, pp. 14–23, Jan. 2011.

[25] JavaTpoint, "Regression vs. classification in machine learning." `https://www.javatpoint.com/regression-vs-classification-in-machine-learning`, Dec. 2023.

[26] D. C. Montgomery, E. A. Peck, and G. G. Vining, *Introduction to linear regression analysis*. Wiley series in probability and statistics, Hoboken, New Jersey: Wiley, fifth edition ed., 2020.

[27] JavaTpoint, "Regression analysis in machine learning." https://www.javatpoint.com/regression-analysis-in-machine-learning, Dec. 2023.

[28] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning: with Applications in R*. Springer texts in statistics, New York NY: Springer, second edition ed., 2021.

[29] B. Kim, R. Khanna, and O. O. Koyejo, "Examples are not enough, learn to criticize! Criticism for Interpretability," in *Advances in Neural Information Processing Systems*, vol. 29, Curran Associates, Inc., 2016.

[30] D. V. Carvalho, E. M. Pereira, and J. S. Cardoso, "Machine Learning Interpretability: A Survey on Methods and Metrics," *Electronics*, vol. 8, p. 832, Aug. 2019. Number: 8 Publisher: Multidisciplinary Digital Publishing Institute.

[31] C. Molnar, *Interpretable machine learning: a guide for making black box models explainable*. Munich, Germany: Christoph Molnar, second edition ed., 2022.

[32] G. Ciatto, M. I. Schumacher, A. Omicini, and D. Calvaresi, "Agent-Based Explanations in AI: Towards an Abstract Framework," in *Explainable, Transparent Autonomous Agents and Multi-Agent Systems* (D. Calvaresi, A. Najjar, M. Winikoff, and K. Främling, eds.), vol. 12175, pp. 3–20, Cham: Springer International Publishing, 2020. Series Title: Lecture Notes in Computer Science.

[33] T. Hastie and R. Tibshirani, "Generalized Additive Models: Some Applications," *Journal of the American Statistical Association*, vol. 82, pp. 371–386, June 1987.

[34] K. Ravindra, P. Rattan, S. Mor, and A. N. Aggarwal, "Generalized additive models: Building evidence of air pollution, climate change and human health," *Environment International*, vol. 132, p. 104987, Nov. 2019.

[35] D. Servén, C. Brummitt, H. Abedi, and Hlink, "dswah/pyGAM: v0.8.0." https://zenodo.org/record/1208723, Oct. 2018.

[36] E. Yaman and A. Subasi, "Comparison of Bagging and Boosting Ensemble Machine Learning Methods for Automated EMG Signal Classification," *BioMed Research International*, vol. 2019, pp. 1–13, Oct. 2019.

[37] JavaTpoint, "Random forest algorithm." https://www.javatpoint.com/machine-learning-random-forest-algorithm, Dec. 2023.

[38] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, (San Francisco California USA), pp. 785–794, ACM, Aug. 2016.

[39] A. Ogunleye and Q.-G. Wang, "XGBoost Model for Chronic Kidney Disease Diagnosis," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 17, pp. 2131–2140, Nov. 2020.

[40] P. Liashchynskyi and P. Liashchynskyi, "Grid Search, Random Search, Genetic Algorithm: A Big Comparison for NAS," Dec. 2019. arXiv:1912.06059.

[41] A. Colin Cameron and F. A. Windmeijer, "An R-squared measure of goodness of fit for some common nonlinear regression models," *Journal of Econometrics*, vol. 77, pp. 329–342, Apr. 1997.

[42] T. Chai and R. R. Draxler, "Root mean square error (RMSE) or mean absolute error (MAE)?," preprint, Numerical Methods, Feb. 2014.

[43] L. Janson, W. Fithian, and T. J. Hastie, "Effective degrees of freedom: a flawed metaphor," *Biometrika*, vol. 102, pp. 479–485, June 2015.

[44] M. Grandini, E. Bagli, and G. Visani, "Metrics for Multi-Class Classification: an Overview," Aug. 2020. arXiv:2008.05756.

[45] H. Teimoorinia, R. D. Toyonaga, S. Fabbro, and C. Bottrell, "Comparison of Multi-class and Binary Classification Machine Learning Models in Identifying Strong Gravitational Lenses," *Publications of the Astronomical Society of the Pacific*, vol. 132, p. 044501, Mar. 2020. Publisher: The Astronomical Society of the Pacific.

[46] V. Ganganwar, "An overview of classification algorithms for imbalanced datasets," *International Journal of Emerging Technology and Advanced Engineering*, vol. 2, no. 4, pp. 42–47, 2012.

[47] F. Gustafsson, "Determining the initial states in forward-backward filtering," *IEEE Transactions on Signal Processing*, vol. 44, pp. 988–992, Apr. 1996.

[48] G. Farnebäck, "Two-Frame Motion Estimation Based on Polynomial Expansion," in *Image Analysis* (G. Goos, J. Hartmanis, J. Van Leeuwen, J. Bigun, and T. Gustavsson, eds.), vol. 2749, pp. 363–370, Berlin, Heidelberg: Springer Berlin Heidelberg, 2003. Series Title: Lecture Notes in Computer Science.

[49] C. Saranya and G. Manikandan, "A study on normalization techniques for privacy preserving data mining," *International Journal of Engineering and Technology (IJET)*, vol. 5, no. 3, pp. 2701–2704, 2013.

[50] H. Henderi, T. Wahyuningsih, and E. Rahwanto, "Comparison of Min-Max normalization and Z-Score Normalization in the K-nearest neighbor (kNN) Algorithm to Test the Accuracy of Types of Breast Cancer," *International Journal of Informatics and Information Systems*, vol. 4, pp. 13–20, Mar. 2021. Number: 1.

[51] T. U. Team, "What is data cleaning? basics and examples." `https://www.upwork.com/resources/data-cleaning-basics`, Jan. 2024.

[52] "Comparing anomaly detection algorithms for outlier detection on toy datasets." `https://scikit-learn.org/stable/auto_examples/miscellaneous/plot_anomaly_comparison.html#sphx-glr-auto-examples-miscellaneous-plot-anomaly-comparison-py`, Jan. 2024.

[53] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation Forest," in *2008 Eighth IEEE International Conference on Data Mining*, (Pisa, Italy), pp. 413–422, IEEE, Dec. 2008.

[54] J. Brownlee, "Recursive feature elimination (rfe) for feature selection in python." `https://machinelearningmastery.com/rfe-feature-selection-in-python/`, Dec. 2023.

[55] P. Everton Gomede, "Recursive feature elimination: A powerful technique for feature selection in machine learning." `https://medium.com/@evertongomede/recursive-feature-elimination-a-powerful-technique-for-feature-selection\-in-machine-learning-89b3c2f3c26a`, Dec. 2023.

[56] "sklearn.feature_selection.rfe." https://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.RFE.html, Jan. 2024.

[57] "Random forest feature importance computed in 3 ways with python." https://mljar.com/blog/feature-importance-in-random-forest, Dec. 2023.

[58] B. H. Menze, B. M. Kelm, R. Masuch, U. Himmelreich, P. Bachert, W. Petrich, and F. A. Hamprecht, "A comparison of random forest and its Gini importance with standard chemometric methods for the feature selection and classification of spectral data," *BMC Bioinformatics*, vol. 10, p. 213, July 2009.

[59] I. Tougui, A. Jilbab, and J. E. Mhamdi, "Impact of the Choice of Cross-Validation Techniques on the Results of Machine Learning-Based Diagnostic Applications," *Healthcare Informatics Research*, vol. 27, pp. 189–199, July 2021.

[60] RITHP, "The main parameters in XGBoost and their effects on model performance." https://medium.com/@rithpansanga/the-main-parameters-in-xgboost-and-their-effects-on-model-performance\-4f9833cac7c, Jan. 2023.

[61] K. Singh, "How to improve class imbalance using class weights in machine learning?." https://www.analyticsvidhya.com/blog/2020/10/improve-class-imbalance-class-weights/, Feb. 2024.

[62] A. Anand, G. Pugalenthi, G. B. Fogel, and P. N. Suganthan, "An approach for classification of highly imbalanced data using weighting and undersampling," *Amino Acids*, vol. 39, pp. 1385–1391, Nov. 2010.

[63] "sklearn.linear_model.logisticregression." https://scikit-learn.org/stable/modules/generatedsklearn.linear_model.LogisticRegression.html, Feb. 2024.

[64] "sklearn.ensemble.randomforestclassifier." https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html, Feb. 2024.

[65] D. F. Hamilton, M. Ghert, and A. H. R. W. Simpson, "Interpreting regression models in clinical outcome studies," *Bone & Joint Research*, vol. 4, pp. 152–153, Sept. 2015.

# Appendix A

## List of Abbreviations

| Symbol | Meaning |
|---|---|
| $AC$ | Alternating current |
| $AIC$ | Akaike information criterion |
| $CRS$ | Cardiac-related signal |
| $CT$ | Computed Tomography |
| $CV$ | Cross-Validation |
| $CVC$ | Cross venous catheter |
| $GCV$ | Generalized Cross-Validation |
| $DoF$ | Degree of Freedom |
| $EIT$ | Electrical Impedance Tomography |
| $FEM$ | Finite Element Method |
| $GAM$ | Generalized Additive Model |
| $IBS$ | Indicator Bolus Signal |
| $LoA$ | Limits of Agreement |
| $LogReg$ | Logistic Regression |
| $LOSOCV$ | Leave-One-Subject-Out Cross-Validation |
| $MAE$ | Mean absolute error |
| $MSE$ | Mean squared error |
| $N/A$ | Not applicable |
| $NaN$ | Not a Number |
| $RF$ | Random Forest |
| $RFE$ | Recursive Feature Elimination |
| $ROI$ | Region of interest |
| $RMSE$ | Root mean sqaured error |
| $VRS$ | Ventilation-related signal |
| $XGBoost$ | Extreme Gradient Boosting |