Silesian University of Technology
Faculty of Automatic Control, Electronics and Computer Science
Institute of Informatics

Doctor of Biomedical Engineering Dissertation

Machine learning-based workflow for the analysis of MALDI-TOF mass spectrometry cancer data

mgr inż. Wojciech Sikora

Supervisor: Prof dr hab. inż. Joanna Polańska

Contents

Acknowledgments

I would like to thank my supervisor, prof dr hab. inż. Joanna Polańska for her guidance, understanding and patience. I would also like to thank my family and friends who supported me with words of encouragement.

This work was co-financed by the European Union through the European Social Fund (grant number: POWR.03.02.00-00-I029).

1. Introduction

Researchers in the field of biology are always striving to expand our knowledge of the processes that occur in living organisms. A deeper understanding of our biology and the biology of other organisms can significantly improve the quality of our lives and help fight diseases. In the 20th century, on the shoulders of genetics and molecular biology, a new field of study, the "omics" sciences, began. These sciences are large-scale studies of organisms that aim to study and quantify the entire process of gene expression from DNA to the biological phenotype of organisms (see Figure 1.1) and the effects of various processes, such as diseases and drug treatments, on this expression. The general idea is that a complex system can be understood more thoroughly if considered as a whole [1].

Proteomics is one of the "omics" sciences, it studies the protein composition of cells, tissues, and even entire organisms. Proteomic research became the viral source of knowledge about organisms, helping us to better understand the information encoded in genomes. The biggest challenge in proteome research is to find out as precisely as possible which proteins are present in the organism under study and in what amounts. This is a very difficult task, but great advancements in mass spectrometry (MS) have made it possible. The discovery of the soft ionization method, for which John B. Fenn and Koichi Tanaka were awarded the Nobel Prize in Chemistry in 2002, was particularly important for proteomics research.

Mass spectrometry is a tool that can be used to study the protein composition of biological samples. With the help of MS, we can identify and quantify proteins in the analyzed mixture [2]. Usually, there are three distinct steps: ionization, mass analysis and ion detection. During the ionization step, molecules of the analyzed mixture are given an electric charge. Ionization makes it possible to separate molecules during mass analysis based on their mass-to-charge ratio (m/z). In the final step, the molecules are detected and counted, giving us the product of MS analysis, the mass spectrum (see Figure 1.2).

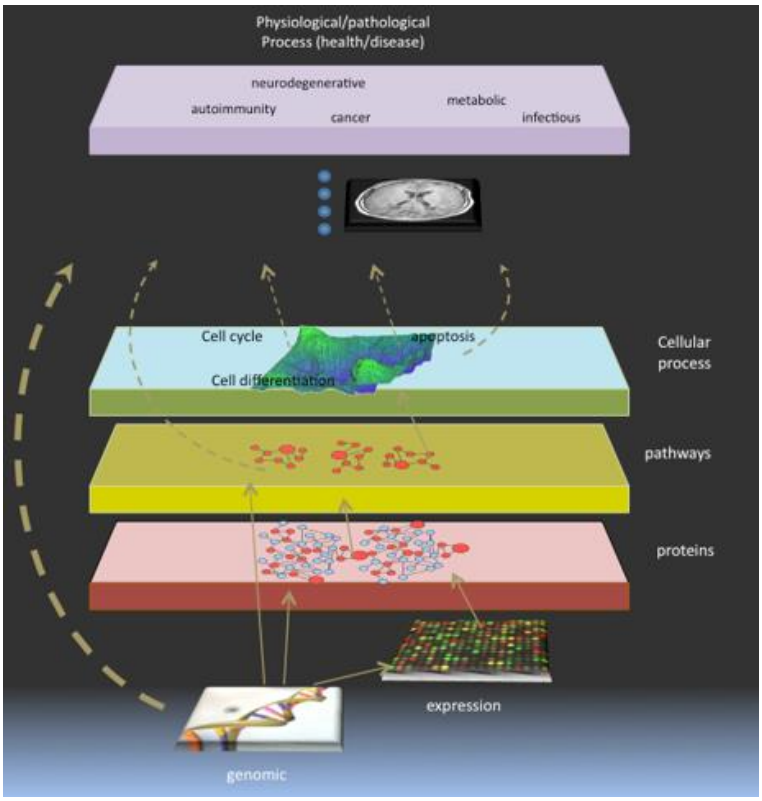

Figure 1.1: Schematic visualization of gene expression from DNA to a disease.
Source: https://baranzinilab.ucsf.edu/data-science.

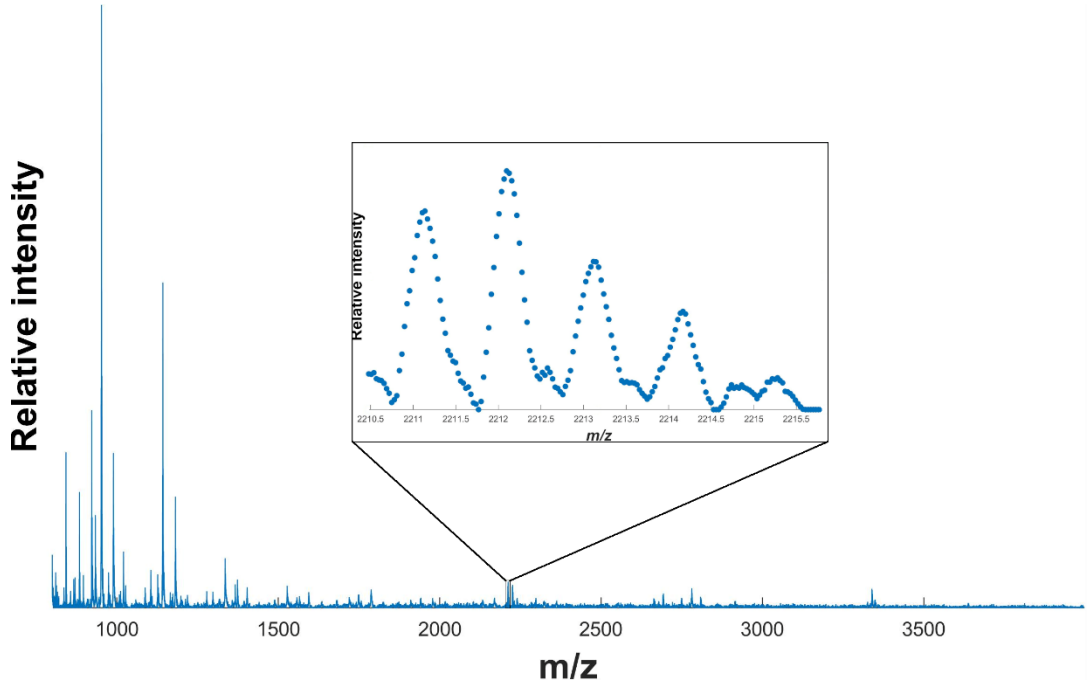

Figure 1.2: An example of the mass spectrum (after baseline correction).

The various types of mass spectrometry differ in the way they analyze protein mixtures. Each technique has its advantages and disadvantages. The main indicators of mass spectrometry performance are molecular resolution, mass accuracy, sensitivity, range and throughput. Molecular resolution is the ability to separate ions with similar *m/z* values. Mass accuracy determines how precisely the mass spectrometer measures the mass of ions. Sensitivity is the ability to detect ions of low intensity. The range is the minimum and the maximum *m/z* values that can be measured, and the throughput describes the speed of the analysis [3].

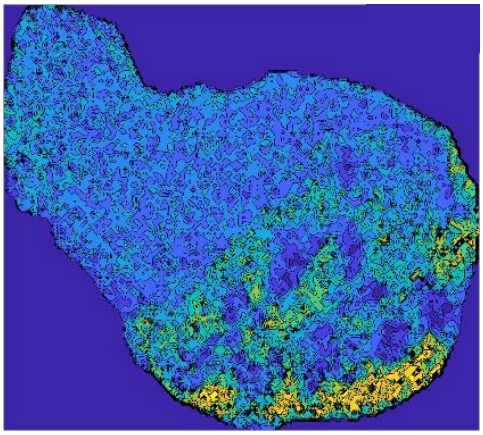

Figure 1.3: An example of an image acquired by MSI. The yellow color marks regions where the molecule has the highest intensity.

Mass spectrometry imaging is another important tool for proteomics research. Mass spectrometry imaging (MSI) adds a physical dimension to the data by defining an (x, y) grid over the surface of the sample. MS data are collected for each pixel of the grid, and images (heat maps) of the individual *m/z* values can be generated, as shown in Figure 1.3 [4]. Obtaining information about the spatial distribution of molecules greatly improves the ability to investigate processes like diseases and drug treatments. In 1997 Richard Caprioli published a paper [5] on the first steps in this field. Since then, many improvements have made the MSI an established tool in biological research, clinical practice, pharmaceutical industry and other fields [4].

Great efforts are constantly being made to improve the performance of mass spectrometry and mass spectrometry imaging. Improvements are being made for all stages of the analysis, sample preparation, ionization methods, mass analysis techniques and detection methods. In general, the focus is to improve two areas, the molecular resolution of mass spectrometry and the spatial resolution of mass spectrometry imaging for both bulk and single-cell analysis, although other aspects

like range and throughput are also important. Some mass spectrometers can identify and quantify proteins with great precision but can only detect a narrow *m/z* range at a time. On the other hand, some methods allow for the analysis of a very wide *m/z* range but result in lower-quality mass spectra. Data quality is very much dependent on the data acquisition parameters (scanning time or dwell time for non-scanning instruments). Thus, usually, trade-offs have to be made between data quality and throughput [3]. For our purposes, the desired traits of mass spectrometry are high resolution, wide range and high throughput.

An obstacle in the meaningful analysis of the MS data is the noise in mass spectra. The noise obscures small intensity signals in measurements and can produce fake peaks. Usually, the noise is dealt with by some kind of filtration, but this also leads to information loss. Noise in data from MS occurs in several forms, including systemic background noise called the baseline, high-frequency noise due to interference with an ion source, noise due to sample contamination, and other types of noise that depend on the type of mass spectrometry [6, 7]. In this work, we are interested in high-throughput mass spectrometry, which can analyze very complex mixtures of proteins with a wide range of *m/z* values. The quality of the data generated by such methods is far from ideal. Such mass spectra consist of Gaussian-shaped peaks around the actual *m/z* value of molecules they represent and are surrounded by noise. Information about protein localization and abundance must be extracted by careful and thorough analysis.

During knowledge discovery in mass spectrometry data, understanding the nature of the data is most important. The initial dimensionality of tens of thousands or even up to a million mass channels is far too large for the application of most data mining methods without prior dimensionality reduction. The preprocessing of the data must be done with field-specific knowledge about mass spectrometry. A typical workflow involves raw data access, baseline correction, peak picking and selection, mass alignment, signal normalization, and molecular annotation [8]. Only then are the traditional steps of feature engineering and machine learning applied, e.g., genetic algorithms [9], neural networks [10], linear discriminant analysis [11], simulated annealing algorithm [12], support vector machine [13], k-means [14, 15] etc.

Imaging adds another layer of complexity to the analysis of MS data, making the analysis a complex and multistep process. Many publications focus on a particular step of the analysis like baseline correction, peak detection or classification of images. Other publications are per case study, customized for a singular application and hard to apply for different data sets. There is a need for a set of tools that enable a comprehensive, automated and data-driven analysis of MSI data, a detailed

workflow that takes raw data and produces a set of meaningful features that can be used with the help of machine learning to train high-performing classifiers. Studies are being done in pursuit of this goal, but the topic is yet insufficiently explored. In particular, there is no workflow that utilizes the imaging information gained form MSI to enhance the feature engineering on the MS data. Proposing such a workflow is one of our goals.

1.1. Goals

The main goal of this dissertation is to prepare a robust, data-driven and detailed workflow for processing mass spectrometry imaging data obtained from biological samples taken from patients with cancer. The result of such processing should be a small non-redundant set of features related to specific peaks in the mass spectra that can be used for training of classifiers and for biomarker discovery. The goal is to prepare a workflow that can be used for any data set acquired with MALDI-TOF mass spectrometry imaging.

The second goal of our work is to apply machine learning on the acquired feature set and examine the quality of our data processing. The goal is to prove that the processing retained the information and hidden patterns in the data and it is possible to use it to train well performing classifiers.

We are also trying to improve on existing methods used during the multi step workflow we propose. In particular we are trying to improve the process of spectrum modelling during the peak detection step and compare it with other state-of-the-art peak detection methods.

The motivation behind this research is that there seems to be a lack of an approach to feature extraction and dimensionality reduction of MALDI-TOF MSI data that simultaneously uses both protein composition information from mass spectrometry and spatial distribution information from imaging. This inclined us to develop a method that uses spatial distribution to help with processing of MS data.

There are also only a few works [16, 17] that describe a fully data-driven approach to the analysis of MALDI-TOF MSI data, and this field is still insufficiently investigated.

The aim is also to propose a workflow that provides a set of features that can be used for protein identification and quantification as well, and, subsequently, for the preparation of a clinical trial.

We state the following. Thesis 1. Peak detection method based on spectrum modelling with Gaussian mixture models with prior mass spectrum segmentation is able to identify peaks in complex mixtures and provide information about peak location as well as its relative intensity.

Thesis 2. Using statistical tests for comparing the spatial distribution is a good approach for redundancy removal and dimensionality reduction of the data, including isotope envelope detection.

Thesis 3. Reasoning about feature importance by aggregating results from many unit models enables to determine feature importance in heterogeneous data, and to find all important features even highly correlated with each other.

1.2. Contributions

The thesis includes several contributions to the field of mass spectrometry imaging data analysis. First, we prepared a complete process of MALDI-TOF MSI data analysis with detailed steps of peak detection, noise filtering and feature engineering that ultimately leads to a well-defined and non-redundant set of features that can be used to train well-performing classifiers.

The second contribution is the examination of the process of spectrum division for modeling individual parts during spectrum modeling [18] and the proposition of indicators to specify the aims of this process and the proposition of a new method for the division.

The third contribution is the novel application of spatial distribution-based decision-making during the feature engineering of the MSI data and during the isotope envelope detection process.

1.3. Organization of the thesis

This thesis is structured as follows.

Chapter 2 provides background information about mass spectrometry and mass spectrometry imaging. The chapter briefly explains why the analysis of imaging data of biological samples is important and introduces the main concepts related to this topic. It also provides important insights into the techniques, most commonly used for this purpose. The information in this chapter helps to understand the challenges a data

analyst faces when working with MSI data by explaining the process of acquiring mass spectra and the differences between mass spectrometry techniques.

Chapter 3 provides a detailed description of the data processed during the experiments. It contains a step by step description of sample preparation and data acquisition, along with all the important parameters and the specification of the used mass spectrometer. This chapter also describes the initial steps taken to prepare the raw data for peak identification.

Chapter 4 is the introduction to the main topic of mass spectrometry data analysis, peak identification. It is an overview of the state-of-the-art peak detection methods for mass spectrometry data. In this chapter we use our data to find peaks using popular methods and compare the results.

Chapter 5 is an in detail description of a custom method for peak detection by spectrum modeling based on Gaussian mixtures. The chapter starts with the inspection of various methods for the division of the spectrum into smaller fragments. It then describes the process of fitting Gaussian mixture models to the fragments with a detailed description of the algorithm and the implementation used to fit the model. Important part of the chapter is the process of choosing the optimal number of mixture elements. At the end of the chapter, we present the results of applying this peak detection method to our data.

Chapter 6 chapter is devoted to feature engineering, with the goal of reducing the dimensionality of the data and removing redundancy. It is a description of the entire process of dimensionality reduction from the set of thousands of Gaussian components of the spectrum model to a small set of features using real-life knowledge about the process of MSI data acquisition. The chapter contains one of the key concepts of the thesis, that is the usage of spatial distribution of features to facilitate the feature engineering process. In detail description of the used methods and the entire feature engineering process is provided.

Chapter 7 describes the application of statistical and machine learning methods to train classifiers capable of making a prediction for a new observation based on the processed data. It contains the strategy for data set division into training, testing and validations sets as well as extensive set of tools for model comparison and performance evaluation. Chapter then describes algorithms used to train classifiers. The algorithms are, multinominal logistic regression-based algorithm and neural networks. This is followed by the comparison of classification performance achieved by both algorithms and the in detail description of the feature importance evaluation.

The final chapter contains discussion about the conducted experiments, drawn conclusions, and plans for future research.

2. Basics of Mass Spectrometry Imaging

MSI experiments generate hundreds of thousands of mass spectra. For each mass spectrum, the number of different *m/z* values can reach millions, depending on the type of MS. Processing such large amounts of data is not easy. One approach might be to treat mass spectrometry as a black box and consider only the output, treating each *m/z* as a separate feature. The downside of this approach is that the real features, are the molecules present in the analyzed mixture, and although there are types of mass spectrometry with very high resolution, a single molecule is usually described by many successive values in the mass spectrum. Also, the large number of features is a problem because most machine learning algorithms cannot train models on data sets with millions of features in a reasonable amount of time. Very extensive feature selection is then required to provide a manageable set of features for classification.

A better solution is to process the spectral data using field-specific knowledge to reduce the dimensionality. In this way, the information about the analyzed mixture is preserved, and at the same time we can remove the noise and address other problems within the data. Therefore, an understanding of the fundamentals of mass spectrometry and mass spectrometry imaging is necessary for effective analysis of MSI data.

2.1. Mass spectrometry

The first mass spectrometers were built in the late 19th century. Initially with very limited applications, but over the years they became complex and sophisticated instruments used in many different fields of science [19]. In simple terms, a mass spectrometer (mass analyzer) separates ions according to their mass-to-charge ratio, although mass spectrometry is usually referred to as the entire process that begins with the ionization of the sample and ends with the acquisition of the mass spectrum. There are many types of mass spectrometry, which differ in the methods of ionization, ion separation and detection.

The first step in mass spectrometry is ionization of the sample. In this step, the molecules of the analyzed substance are electrically charged. This can be done, for example, by attaching an additional proton to the molecule or by removing an electron from the molecule. In the next step, the ions are directed into the mass spectrometer, where they are separated according to their *m/z* value. This is done based on the physical properties of the ions in the electric or magnetic field. Finally, during the detection step, the charge induced by the ions or generated by their current is measured. The result is a function of the relative abundance and mass-to-charge ratio of the ions, called the mass spectrum.

As mentioned earlier, mass spectrometry is used in many different fields, but it's only relatively recently that it has been used for the analysis of biological samples. Thanks to advances in the performance of MS and especially to the invention of soft ionization methods, MS has become an irreplaceable technique in the analysis of biologically related molecules [1, 20].

2.2. Mass spectrometry imaging

The goal of proteomics is not only to map all proteins in an organism, but also to measure and link protein expression to specific processes. Another interest of proteomics research is to study the movement of proteins, the rates of their production and degradation, the interaction between them and others [21]. To be able to study such complex issues, simple inspection of protein composition is not enough. What can be helpful is information about the physical localization of proteins and their concentrations in different regions of a cell or tissue. This information can be obtained by mass spectrometry imaging.

The first step in the field of MSI was taken by Richard Caprioli in 1997, as described in his paper [5]. Since then, the number of publications on this topic has been increasing every year with reports of new applications and improvements in both apparatus and data analysis techniques. Application of this technique to biological samples provides information about the spatial distribution of peptides and proteins, which in turn can provide valuable insights about the organism state. For example, the presence or absence of certain proteins can be correlated with the pathological condition of a tissue, such as cancer. The combination of information about the protein composition in the tissue and their spatial distribution helps us to better understand the processes taking place in organisms.

Imaging involves dividing the area of the sample into a grid of pixels. Mass spectrometry is then performed for each pixel, adding a new dimension to the MS data. In this way, a heat map (see Figure 2.1) can be created for each ion, showing how a particular molecule is physically distributed on the tissue.

The main feature of MSI is the spatial resolution of the image, which depends on the size of the pixel on the grid. The higher the spatial resolution, the smaller the pixels and the better the quality of the image. The bottleneck for the spatial resolution of the MSI is the ionization of the sample. The minimum size of the pixel is the smallest area that an ionization method can ionize at one time. Ionization methods for MSI are constantly being improved, and a great emphasis is put on improving spatial resolution.

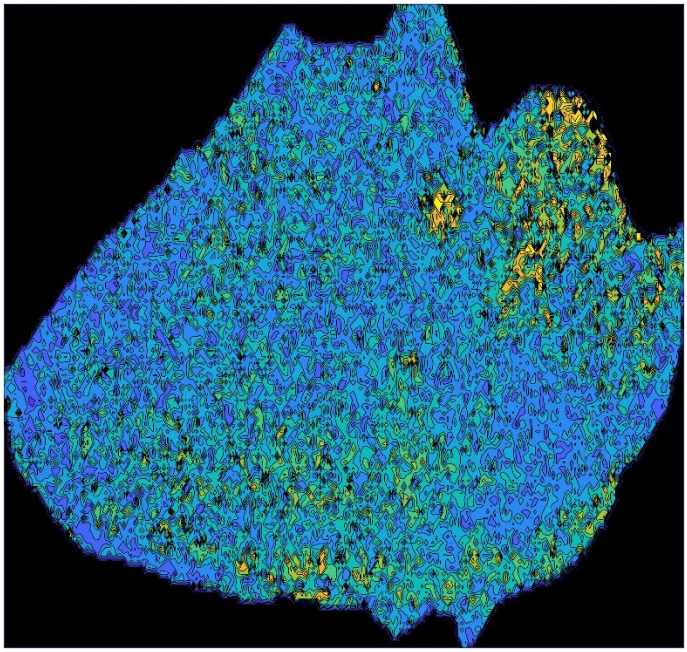

Figure 2.1: An example of a mass spectrometry image. The image shows the spatial distribution of a specific mass-to-charge ratio. Each pixel on a sample is taken form a different mass spectrum.

2.3. Ionization

Ionization is arguably the most important part of mass spectrometry, as it has great impact on the data and the overall characteristics of a mass spectrometry technique. As mentioned earlier, the sole purpose of ionization is to give an electrical charge to the molecules on the sample, creating ions. The challenge is to ionize as many molecules as possible, preferably all with the same charge, and keep them intact during the

process. Ideally, every single molecule on the sample would receive an identical electrical charge, then all ions would be directed to the mass analyzer. However, real ionization methods are not ideal. Each ionization method approaches the task differently and has different characteristics and limitations, therefore, choosing a mass spectrometry type is a per-case decision.

Ionization is especially difficult for biological samples because molecules like proteins and peptides are very fragile. Therefore, one of the most important features of an ionization method for biological samples is the amount of energy used. The amount of energy introduced to the sample categorizes ionization methods into two groups, hard ionization and soft ionization. The difference lies in the excess energy generated during the process. In hard ionization methods, there is greater excess of internal energy of the ions, which leads to the fragmentation of the molecules. In soft ionization methods, smaller amounts of energy are introduced. For biological samples, this means that the chemical bonds in the molecules can remain intact.

Some methods, by definition, work only with samples in a particular physical state, and the first step of the ionization is the conversion to the into liquid or gaseous phase. Apart from the inability to work with solid samples, the problem with such methods is the introduction of an additional medium as a source of electrons in which the sample is dissolved. The used solvent or medium can negatively interact with the molecules of the sample and is the source of noise in the mass spectrum. For MSI, of course, only methods that can ionize solid samples are applicable.

Another important feature of an ionization method is the requirement for a specific environment in which the ionization takes place. Many methods require a vacuum or high vacuum to operate, and for many applications such conditions are unacceptable. Ionization also does not always give molecules the same charge. Multiply charged molecules have the same mass as their singly charged equivalents, but their *m/z* values are different. With such ionization methods, a single molecule is present more then one location in the mass spectrum.

In this work, we are primarily interested in ionization methods that can be used for MSI. In addition, we are interested in studying protein and peptide mixtures, which means that we are interested in soft ionization methods.

Currently, the most commonly used soft ionization techniques for MSI of biological samples are desorption methods like DESI, MALDI and SIMS.

2.2.1. Desorption electrospray ionization - DESI

Electrospray ionization (ESI) is a soft ionization technique for the analysis of liquid samples. Ions are created by passing a solution of sample and solvent through a small capillary. When a voltage is applied to the liquid, the electrostatic field and surface tension of the liquid affect the shape of the liquid at the end of the nozzle. At a certain voltage, the shape becomes a pointed cone (called a Taylor cone), and from the tip of the cone, an aerosol of charged droplets is ejected [22, 23] (see Figure 2.2). These droplets consist of both the analyte and the solvent, and after the droplets leave the nozzle, the solvent evaporates, the ions are released, and are directed into the mass analyzer.

This simple design practically has no limit to the size of ionized molecules, which makes this ionization method good for studying proteins and even protein-protein complexes. However, the sample is constantly consumed, and ions cannot be analyzed continuously by the mass analyzer. Therefore, some of the information is lost and the higher-concentration analytes can suppress the signal from lower-concentration analytes. Finally, since the method requires a liquid form of the sample, it cannot be used for mass spectrometry imaging.

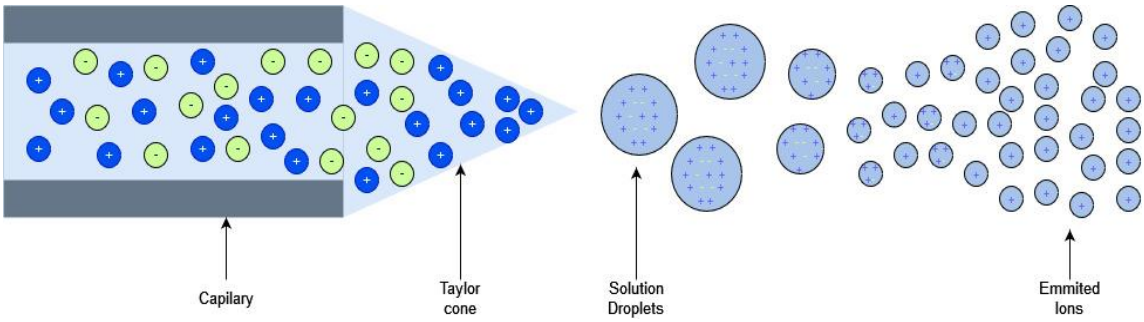

Figure 2.2: Electrospray ionization. The stream of sample and solvent mixture is ionized by an electrostatic field and directed into the mass analyzer.

For mass spectrometry imaging, electrospray ionization was combined with the desorption ionization method in desorption electrospray ionization (DESI). This method was developed in 2004 by R.G. Cooks et al. [24]. In ESI the solution of the sample and solvent is sprayed through the capillary. In contrast, in DESI, only solvent droplets are sprayed. The droplets are the ion source and are sprayed on the sample, which is placed on an insulating surface. The impact of the charged particles on the

surface of the sample produces gaseous ions from the sample molecules. The ions are then directed into the mass analyzer (see Figure 2.3).

Ionization occurs under atmospheric pressure. This is an important advantage of DESI over other ionization techniques used for MSI. Also advantageous is the lack of a sample preparation step, as this eliminates many potential errors and noise in the data. DESI is a high-throughput method, and the results are obtained very quickly, which is especially important for proteomic research. The mass resolution offered by this method is quite good. DESI offers a spatial resolution of 50-200 μm for most of the current studies, with maximal spatial resolution reaching 10-20 μm. [25, 26]. The downside of DESI is that it produces multiply charged ions.

Overall, DESI is an appealing method for MSI analysis, and it is frequently used for proteomics-related studies. The sensitivity and spatial resolution of DESI can be increased, and improvements are constantly made with such methods as, for example, nano-DESI [27] or AFADESI [28].

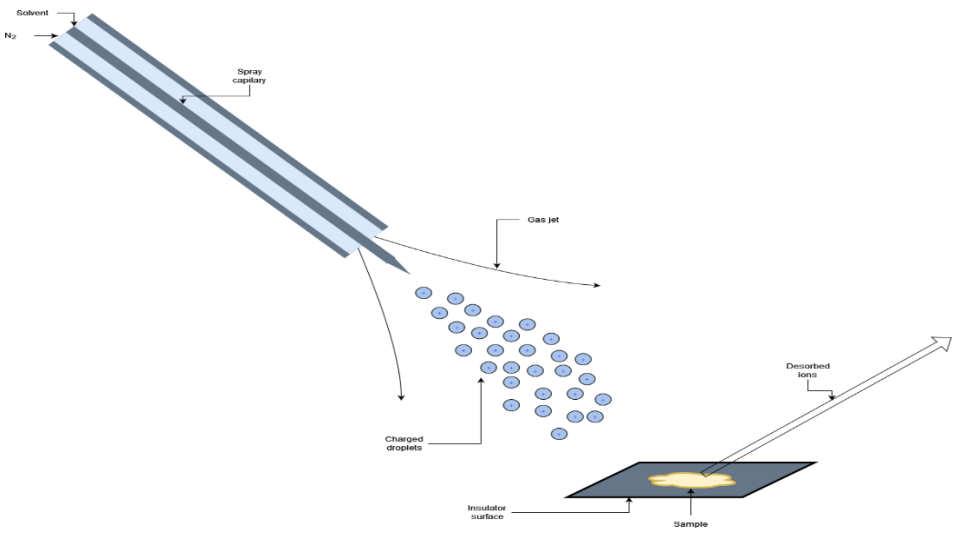

Figure 2.3: Desorption electrospray ionization (DESI). Charged droplets directed at the sample cause desorption and ionization of the sample molecules.

2.2.2. Matrix-assisted laser desorption ionization - MALDI

Matrix-assisted laser desorption ionization (MALDI) is the most widely used ionization technique for imaging of biological samples. As the name implies, ions are generated with the help of a laser that serves as the energy source for desorption. The laser pulse is directed at the sample mixed with the matrix material. The energy emitted by the laser pulse causes the desorption (ejection) of the sample and matrix in

the direction of the mass analyzer. The matrix material vaporizes after desorption and transfers its charge to the sample molecules, creating ions. Figure 2.4 shows the visualization of this process.

The use of the matrix material has two functions. First, the matrix absorbs the energy of the laser and protects the molecules from excess energy, making MALDI a very soft ionization technique. Second, the matrix is the source of protons for ion formation during matrix evaporation.

The presence of the matrix also has disadvantages, as it adds the sample preparation step to the process. The matrix material is also a source of noise in the mass spectrum. DESI doesn't have such disadvantages. Another disadvantage of MALDI over DESI is the need for a vacuum environment.

The biggest advantage of MALDI is that the laser can be directed very precisely on the sample. Therefore MALDI offers a spatial resolution of about ten micrometers, and in some experiments, it reaches about 1.4 μm [26]. Thanks to the use of the laser, this method also has high sensitivity. The laser can be applied with short burst so that very little of the sample is wasted, unlike ESI where the sample is consumed continuously. This, in turn, means that even low-intensity molecules can be detected. Another key feature of MALDI is its high throughput. The ability to generate a large number of mass spectra in small amount of time is critical for obtaining data sets large enough for effective knowledge discovery. The high spatial resolution combined with high throughput, wide *m/z* range and reliable results make MALDI the leading ionization method for MSI of biological samples.

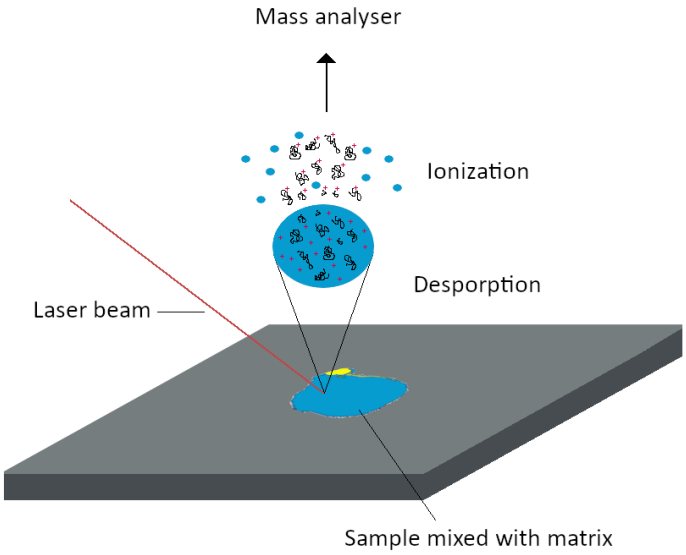

Figure 2.4: Matrix-assisted laser desorption ionization (MALDI). Laser beam is the energy source for desorption of the matrix-sample mixture.

2.2.3. Secondary ion mass spectrometry - SIMS

Secondary ion ionization is another desorption method used for imaging of biological samples. The working principle is very similar to that of the MALDI, but the energy for desorption is provided by a primary ion beam instead of a laser. Ionization is achieved by directing a primary ion beam on the sample. Collision of the ion beam with the molecules of the sample causes sputtering of secondary ions. The beam of secondary ions is then directed at the mass analyzer.

The sample doesn't require prior preparation of any kind. SIMS method doesn't use matrix material because the primary ions are the source of electric charge. This also means that the molecules of the sample are not protected from the excess energy by the matrix and the method is not as soft as MALDI. On the other hand, the ion beam can be focused much more precisely than the laser beam, with a precision up to 50 nm (0.05 μm). This leads to the very high spatial resolution of this method.

The downside is that only a fraction of disrobed molecules are ionized, and the molecules are exposed to much higher energies than in laser or electrospray methods. Crucially, the ionization occurs under an ultrahigh vacuum to avoid collisions between primary and secondary ions. This limits the application of this method to samples that can survive these conditions. SIMS offers very high spatial resolution, and although it can be used with some biological samples, MALDI is still considered a go-to method for proteomic-related studies. However, there are improvements being made to mitigate the problems with SISM and utilize superior spatial resolution. The problem

with excess energy has been addressed by the development of primary ion guns used for MSI, that reduces the fragmentation in molecules [29]. More recent developments are related to the use of gas cluster ion beams, which have increased the applicability of SIMS for analysis of biological samples [30].

2.2.4. Other ionization methods

DESI, SIMS and MALDI remain the most commonly used ionization methods for MSI of biological samples. The majority of publications on the topic of MSI data analysis analyze the data obtained by these methods. There are, however, other ionization techniques and variations of mentioned methods that could be considered better in some aspects or have the potential to be better in the future like LAESI [31], SMALDI [32], IR-MALDI [33], SALDI [34], EASI [35] to mention just some of them.

2.4. Ion separation and detection

Proper ionization is an important part of mass spectrometry, but the ion separation is also complex and important step. Similar to ionization, separation by mass-to-charge ratio is a task that can be accomplished in different ways. In general, the differences in the behaviour of ions of different masses in an electric or magnetic field are used to separate the ions.

The most important characteristics of a mass spectrometer are mass resolution, sensitivity, range, accuracy, and throughput. Mass resolution describes the ability of mass analyzer to separate ions with very similar *m/z* values. Sensitivity describes how well the mass analyzer detects ions with a given mass-to-charge ratio. For example, some mass analyzers have high sensitivity for certain ions and lower sensitivity for the remaining ions. Others have high sensitivity for low *m/z* and low sensitivity for high *m/z*. The range of the mass analyzer is the minimum and maximum *m/z* values that it is capable of separating. Mass accuracy describes how accurately *m/z* value is measured and finally the throughput is the speed of the analysis just like for ionization.

There are various designs for mass analyzers. Sector mass spectrometers, for example, use the electric or magnetic field to bend the trajectories of the ions. The detector distinguishes the *m/z* value of the ions based on how much the ion's path or velocity has changed. A different design is the quadrupole filter mass analyzer. In this

mass spectrometer, ions are passed through two pairs of rods with a specific voltage applied to them. Based on the voltage, only the ions within a specific *m/z* range reach the detector, and others crash into rods and are therefore filtered out. Quadrupole ion traps have similar design, but instead of passing through the quadrupole, the ions are trapped inside, then, changes in the voltage applied to the rods release ions of specific *m/z* to the detector. Another design is the orbitrap, a method that offers high accuracy and sensitivity and a wide range of *m/z* values. Orbitrap traps the ions in a static field and then continuously measures the *m/z* values of trapped ions.

In this work we are processing the data acquired by MALDI time-of-flight (TOF) mass spectrometry. Therefore we are primarily interested in TOF mass analyzer. Time-of-flight mass spectrometry was first proposed by Stephens [36] in 1946. The proposed technique was based on a simple physics principle that ions with different mass-to-charge ratios, but equal energy or momentum, separate in a constant electric field according to their *m/z* values [37]. Different way of putting it is that ions with the same charge have the same kinetic energy and therefore their velocity in an electric field depends only on their mass. Knowing the strength of the accelerating field, the length of the path, and the time of flight of ions, their mass can be calculated. The first experimental instrument using this principle was created by Cameron and Eggers [38], and the first commercial design was created in 1955 by Wiley and McLaren [39] (see Figure 2.5).

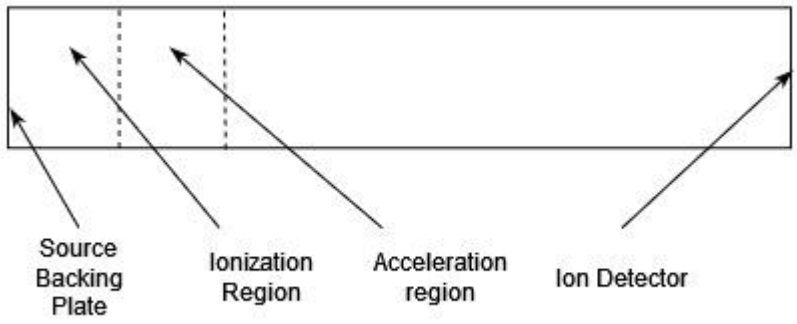

Figure 2.5: Schematic diagram of the early design of the time-of-flight mass spectrometer.

First, a pulsed voltage is applied to the source backing plate in order to form ions. The duration of the pulse is sufficient to remove all ions from the ionization region. The acceleration region has a constant electric field that gives ions their energy. The

ions travel through a field-free drift tube with velocities dependent on their *m/z* ratios and then reach the detector.

At first the performance of TOF mass analyzers wasn't particularly good because of the uncertainty with the initial position of ions entering the mass analyzer. A great improvement in the resolution of TOF mass analyzers was achieved witch the Reflectron (see Figure 2.6). In Reflectron, the path of ions is a curve instead of a straight line. The ions from the source are directed to the electric field that decelerates and then reflects ions to the detector. Ions with the same *m/z* ratios but different kinetic energies due to initial position, have different velocities when entering the decelerating electric field. The ions with higher kinetic energies travel further and, therefore, for a longer time before being reflected. This difference in time of flight compensates for initial differences in kinetic energies of ions with the same *m/z* ratios and results in better performance.

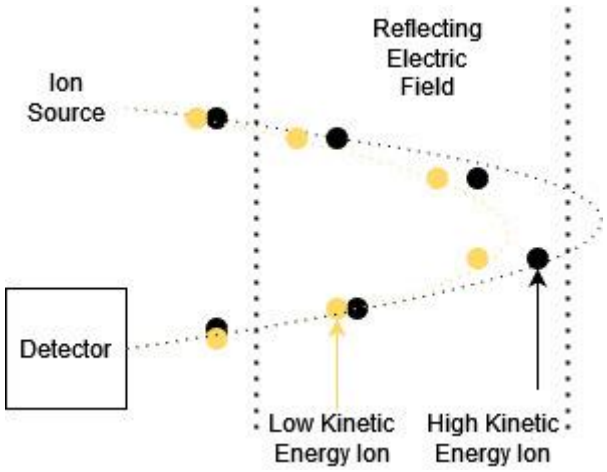

Figure 2.6: Schematic diagram of the Reflectron type of TOF mass analyzer.

Since then, many other designs and improvements have been made to increase the resolution and compatibility of TOF analyzers with various ionization methods, but the working principle remains the same. One of the most important reasons why TOF mass spectrometry is used for imaging is the speed of the analysis. Time-of-flight mass spectrometry is very fast because it can scan the entire mass spectrum at once. Thousands measurements are made during imaging, so the speed is of the essence. There is also no limit to the mass of ions it can separate. TOF mass spectrometers are also used with SIMS and DESI ionization methods.

3. Data acquisition and preprocessing

MALDI has a good balance of properties with good spatial resolution, soft ionization and minimal sample preparation. The time-of-flight mass analyzer, with its high resolution, sensitivity, and ability to process the entire mass range at once, is successfully being used together with MALDI in MALDI-TOF mass spectrometry [40].

There are also other mass analyzers that are paired with MALDI, such as quadrupole mass analyzers, ion trap analyzers, or Fourier transform ion cyclotron resonance [41]. However, the majority of imaging data for clinical research is provided by MALDI-TOF [42].

There are also experiments being made with three-dimensional mass spectrometry for biological samples [43]. The interest in 3D mass spectrometry increases with the number of publications on the topic [17, 44], but the amount of data that needs to be processed is orders of magnitude grater then for two-dimensional imaging. Therefore, most imaging data for clinical research is still two-dimensional and acquired using MALDI-TOF mass spectrometry.

For this reason, the focus of this work is on processing MALDI-TOF MSI data, although the entire workflow or parts of it can be applied to all MSI data.

3.1. Sample preparation and data acquisition

Four patients with head and neck cancer underwent surgery that provided samples evaluated by a specialist pathologist (see Figure 3.1). Samples were then frozen and then 10 μm tissue sections were cut and dried under vacuum. Samples were then washed twice in 70% ethanol and once in 100% ethanol, dried, coated with trypsin solution and incubated for 18 hours at 37 °C in a humid chamber to perform tryptic digestion of proteins.

Tissue sections were imaged using a MALDI-TOF/TOF ultrafleXtreme mass spectrometer (Bruker Daltonik, Bremen) equipped with a smartbeam II™ laser operating at 1 kHz repetition rate. Ions were accelerated at 25 kV with PIE time of 100 ns. Spectra were acquired in positive reflectron mode in the 800–4000 mass range.

A detailed description of the data acquisition steps is given in a publication by Bednarczyk, Gawin et al. [15].

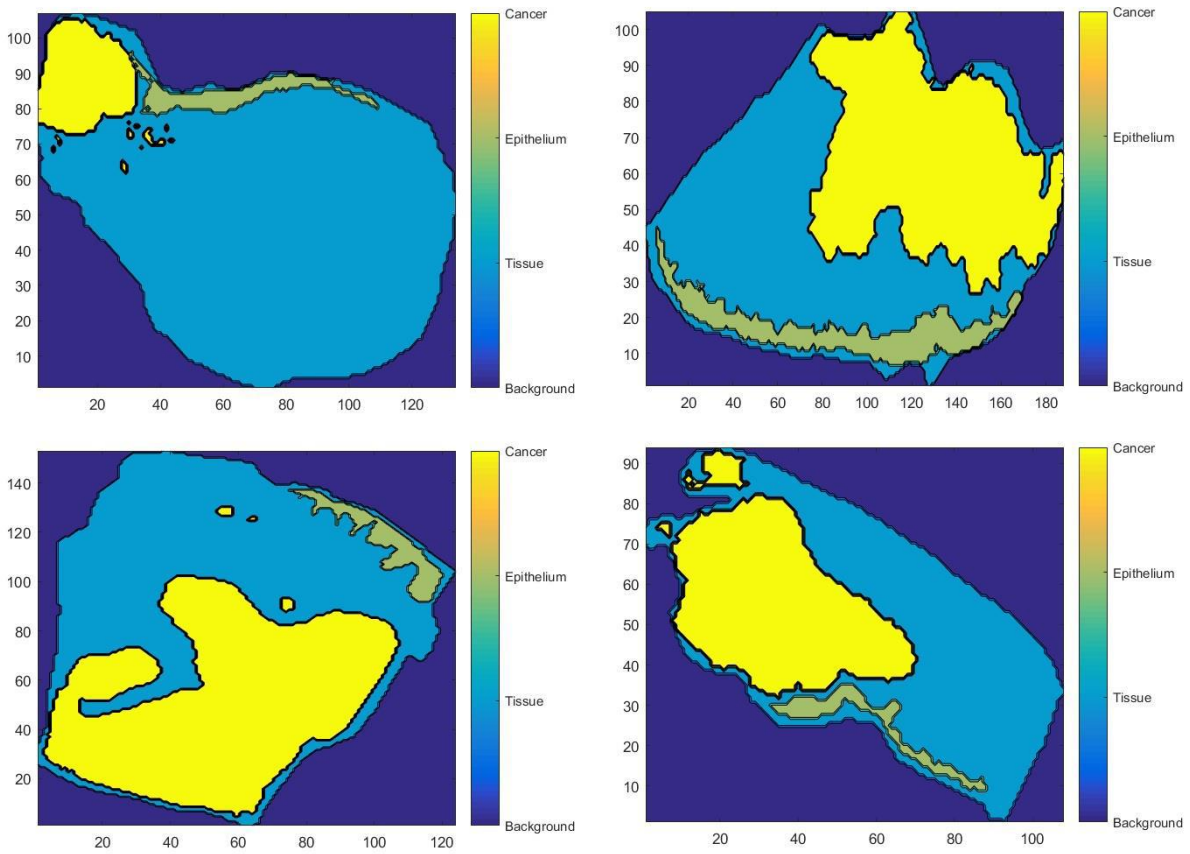

Figure 3.1: Tissue sections from patients with squamous cell carcinoma of the head and neck in which the cancerous and normal epithelial areas were marked by a specialist pathologist.

3.2. Preprocessing

MALDI-TOF MSI can produce gigabytes of data. The four samples produced over 150 thousand mass spectra, each with over 100 thousand mass channels (distinct *m/z* values). The data produced by MALDI-TOF needs a number of transformations before machine learning algorithms can be applied. The reason for this is the nature of MALDI-TOF mass spectrometry. A number of preprocessing steps have been recognized as necessary to control factors that can obscure true differences between

disease classes. Normalization, baseline correction, spectrum alignment and peak detection are among the most common adjustments employed [45].

There are already some tools for MALDI-TOF data processing. Universal tools like Mass-Up [46], MALDIquant [47] or MassSpecWavelet [48] and tools specialized for a specific part of the preprocessing, e.g. SpecAlign [49] for spectrum alignment. There are also general purpose tools for data manipulation and data processing of other types of mass spectrometry, that can also be helpful when working with MALDI-TOF data [50, 51].

Baseline correction is the process of removing the baseline shift caused by the matrix ions reaching the detector at random times [52]. This shift appears in the mass spectrum as a curvature of the signal, especially al low *m/z* values. The baseline shift can be thought of as low-frequency noise in the mass spectrum. Methods used for baseline correction include wavelet-based techniques [53], splines, stochastic Bernstein approximation [52]. In this work we used an approach described in the work of Bednarczyk et al. [54]. It is a modification of the standard cubic spline approach with adaptively adjusting frame width. The Pearson correlation coefficient and appropriate statistical tests were used to examine the trend within the frame. When the frame width is set, 10% signal quantile inside the frame is calculated and cubic spline algorithm is used to obtain the corrected baseline (see Figure 3.2).

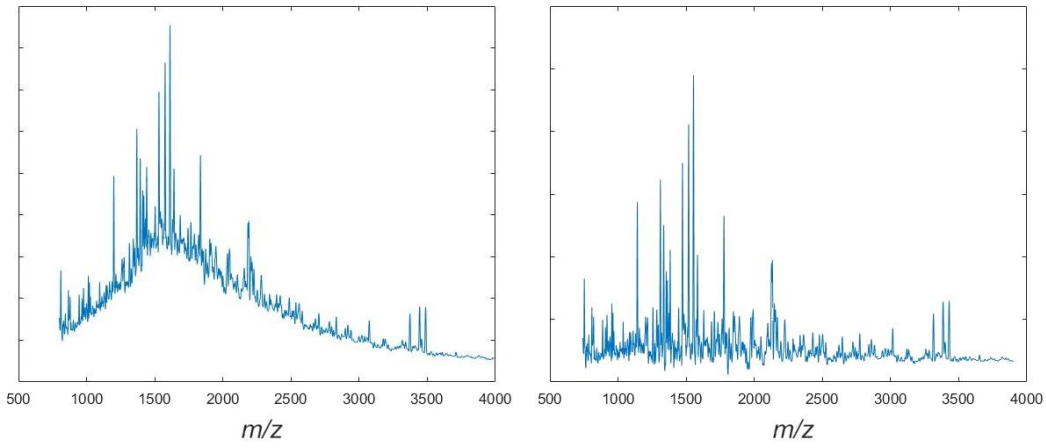

Figure 3.2: Mass spectrum before (left) and after (right) baseline correction.

The next step was to normalize the mass spectra and remove the outliers. Normalization is the scaling of each spectrum for better comparison of intensities between spectra. This is necessary because the mass spectrum is not strictly quantitative, intensities only describe the relative abundance of ions in the context of

a single mass spectrum. In order to compare or aggregate multiple mass spectra, normalization is required. There are a several common methods for mass spectrum normalization. For this work, we used the most common method, called Total Ion Count (TIC) normalization. TIC is the sum of all intensities in the mass spectrum. The normalization is done by dividing each intensity in the mass spectrum by the TIC. In addition, the outlying spectra with too high and too low total ion count were removed using the Bruffaerts's criterion for extremely skewed distributions [55]. Finally all mass spectra were aligned using the Fast Fourier Transform [56].

After the initial processing of the raw mass spectra, the most important part of the MS data analysis can begin, the peak detection. Peak detection aims to extract the real information about the composition of the sample from the noisy mass spectrum. In the next chapters, we will describe and compare the state-of-the art peak detection methods in detail. We will also present a customized peak detection method based on Gaussian mixture spectrum modeling.

4. Methods of peak detection

Preprocessing of data from MS is crucial for successful knowledge discovery. The goal of this process is to identify all peaks in the signal that correspond to a real component in the analyte and convert thousands of data points into a set of features small enough for further analysis. After peak detection, the data can be further analyzed, e.g., for the purpose of biomarker discovery or classification. The mass spectra processing steps mentioned in the previous chapter often precede peak detection, but many peak detection methods have built in mechanisms for baseline correction and other steps such as smoothing and denoising. A good example is a wavelet transform-based approach described by Du Pan et al. [48] where all these steps are done at once. In general, we divide peak detection methods into three groups: peak picking, peak modeling, and spectrum modeling.

4.1. Data aggregation

Peak detection can be performed for a single mass spectrum or for aggregated mass spectra. It might be advantageous to look at each mass spectrum individually, but only simple peak picking methods can do this in a reasonable amount of time considering the number of mass spectra in the data set. With more complex methods that take more time, peak detection for each spectrum is not feasible. Therefore, here we will compare peak detection methods using aggregated representation of all mass spectra in our data set.

Aggregation of mass spectra ensures that all data are used and that computation time does not depend on the number of mass spectra. Aggregation of mass spectra is done by applying aggregate for each *m/z* value across entire population of mass spectra. This requires an additional step of *m/z* unification across mass spectra by binning.

The aggregate, of course, affects the final shape of the signal and thus the outcome of the dimensionality reduction process. Usually the mean spectrum is used, e.g., in [57] the undecimated discrete wavelet transform is used for feature extraction on a mean spectrum. Using the mean as an aggregate results in a smooth signal. However, using such an aggregate can hide meaningful information. Potentially, peaks that are very important to the problem under study may go undetected during peak detection because their averaged intensity in the aggregated signal is below the signal-to-noise ratio. This is especially true for unbalanced data sets, where only a small number of mass spectra contain a particular peak. In such a case, it is even more difficult for that peak to rise above the noise level in the averaged mass spectrum. Since our data is unbalance this might not be the best approach. Figure 4.1 shows how the averaged mass spectrum compares to a single mass spectrum randomly selected from the data set.

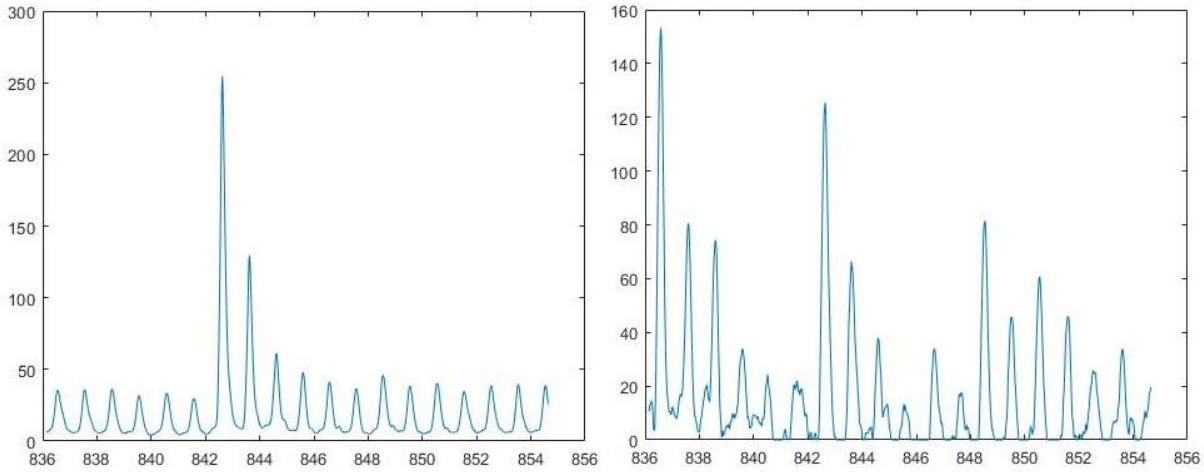

Figure 4.1: Part of the averaged mass spectra (left) and the corresponding part of a random mass spectrum (right).

To solve this problem, another aggregate can be chosen. Using maximum as the aggregate is one option. In this case, class balance in not a problem because the number of mass spectra does not affect the aggregate. The problem with this approach is that the aggregate signal is rough due to imperfect normalization of mass spectra and further analysis is more difficult (see Figure 4.2).

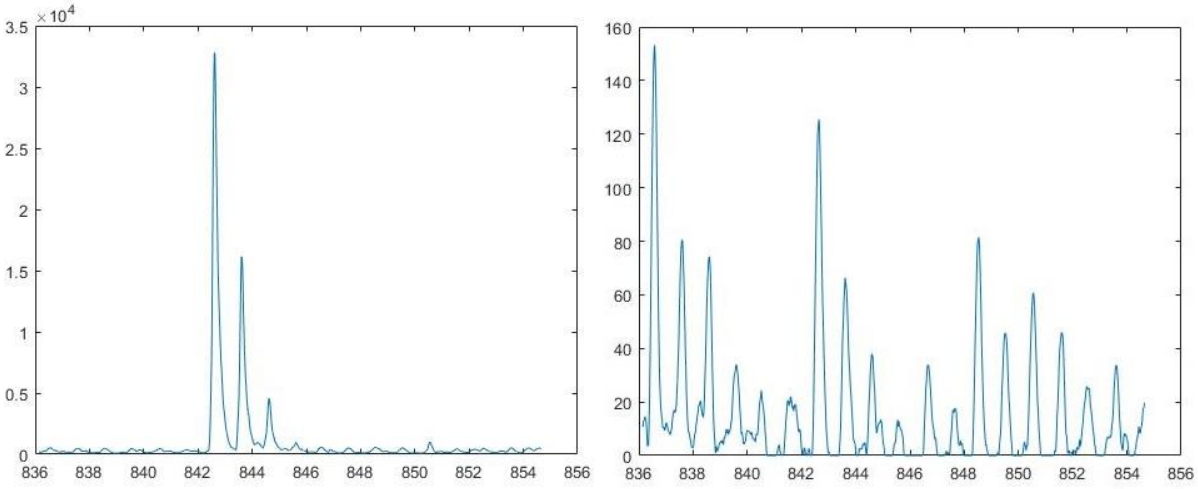

Figure 4.2: Part of the maximum aggregation of mass spectra (left) and the corresponding part of a random mass spectrum (right).

In this work, we used an intermediate solution. The aggregate mass spectrum is determined using the 95th percentile. Although the result is very similar to the mean spectrum, using this aggregate leads to different results. For this method, peak detection is able to find peaks that are obscured in the mean spectrum. Using this approach provides an optimal trade-off between signal smoothness and information loss due to underrepresented peaks being obscured by the noise (see Figure 4.3).

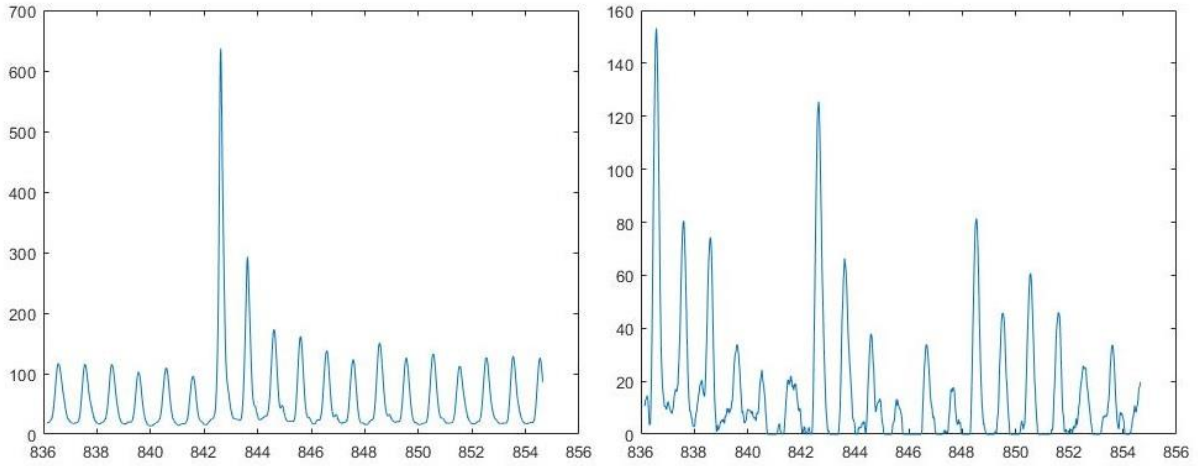

Figure 4.3: Part of the mass spectra aggregated with 95th quantile (left) and the corresponding part of a random mass spectrum (right).

4.2. Peak picking

Pick-picking is the most basic approach to peak detection, usually preceded by baseline correction and smoothing. Methods that fall into this category generally select peaks based on some sort of threshold.

Many such methods are listed in [58]. For example, the threshold can be set for the left and right slopes of the candidate peak. Peak picking can also be done by simply searching for maxima within the local neighborhood. Regardless of the measure for which the threshold is set, its value is calculated based on the definition of the noise. Therefore, the common feature of peak picking methods is their inability to detect low-intensity peaks obscured by the noise, and their sensitivity to the definition of noise and threshold. Figure 4.4 shows an example of the simplest peak picking method with a globally set threshold for peak intensity. On the left we see a raw mass spectrum and on the right the processed spectrum with marked peaks.

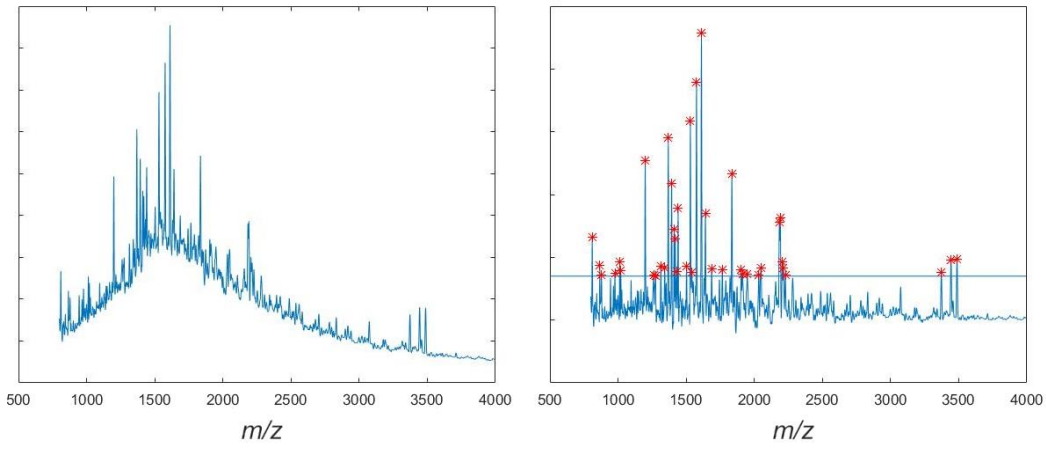

Figure 4.4: Raw mass spectrum (left) and the spectrum after baseline correction and peak picking with signal-to-noise ratio-based global threshold (right).

The most commonly used peak picking method identifies peaks by searching for local maxima with a threshold based on the local signal-to-noise ratio (SNR) [48]. The most important part of this method is, of course, the definition of noise. In statistics, noise can be defined as the median absolute deviation (MAD). In signal processing, it can be defined as the estimated background [59]. A threshold for SNR determines the sensitivity to low intensity peaks. In general, the SNR above 3 is considered minimally acceptable, and the SNR of 10 and above is considered good [60]. A good signal-to-noise ratio means that the signal is distinguishable from noise. For peak identification,

this means that the detected peaks are most likely not correlated with noise. The lower the value of the threshold, the less conservative the identification of peaks, which in turn leads to the increased false positive rate (identification of false peaks). We defined the SNR as:

$$SNR = \frac{f}{MAD}, \tag{1}$$

where f is the peak intensity, and MAD is the mean absolute deviation.

The MAD and the SNR for each point in the signal are calculated using a local window of a given width. Figure 4.5 shows the peaks detected in the aggregate spectrum. The method was applied for two sizes of the window for which the SNR is calculated and for different peak detection thresholds. The figure illustrates the significant impact of parameters values on the outcome of peak detection. As the threshold value decreases, more and more peaks are detected and the problem of false peaks increases, especially for high *m/z* values.

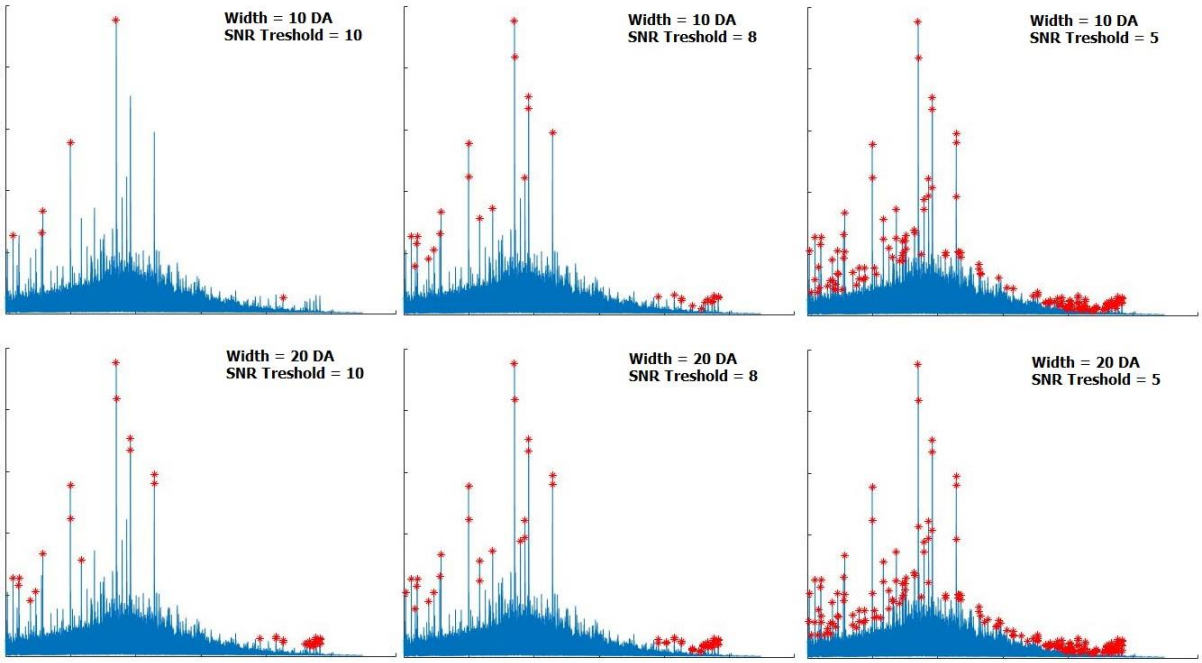

Figure 4.5: Peaks found in the signal using different thresholds and local window widths.

Peak picking is a simple, fast, has a small number of parameters and is easy to implement. On the other hand, the outcome depends heavily on the definition of the noise and values selected for parameters. Overall, peak picking is a good choice for

simple experiments and for the analysis of low complexity mixtures or when the detection of low intensity peaks is not a priority. For more complex mass spectrometry data, effective peak detection can take into account the shape of the peaks. This is because true peaks have a specific shape, unlike peaks associated with noise. Methods that take this into account are categorized as peak modeling.

4.3. Peak modeling

Peak modeling is a more complex method for peak detection. This approach assumes that true peaks have a specific shape. One method is to use the shape ratio of the peak, understood as the area under the curve around the peak candidate divided by the maximum of the entire peak population. The shape ratio is used as the threshold for peak picking. Another example is the continuous wavelet-based approach described in [48]. This method uses continuous wavelet transform (CWT), to transform the signal into wavelet space, and then searches this space for ridge lines that mark the peaks in the signal. CWT is a good method for pattern matching, it is mathematically represented by equation 2 [61].

$$C(a,b) = \int s(t)\psi_{a,b}(t)dt, \tag{2}$$

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}}(t)\psi\left(\frac{t-b}{a}\right), a \in \mathbb{R}^{+}\backslash\{0\}, b \in \mathbb{R}, \tag{3}$$

where $s(t)$ is the signal, a is the scale, b is translation, $\psi(t)$ is the mother wavelet, $\psi_{a,b}(t)$ is the scaled and translated wavelet and C is the two-dimensional matrix of coefficients.

The values in the coefficient matrix indicate how well the signal matches the shape of the wavelet. The first dimension of the matrix corresponds to the translation, and its length is equal to the length of the signal. The second dimension of the matrix corresponds to the scale. Each row in the coefficient matrix describes a single scale of the wavelet. High values imply a better correlation between the signal and the wavelet shape at the given location for the given scale of the wavelet.

The mother wavelet used in the publication by Du et al. [48] is the Mexican Hat Wavelet (see Figure 4.6).

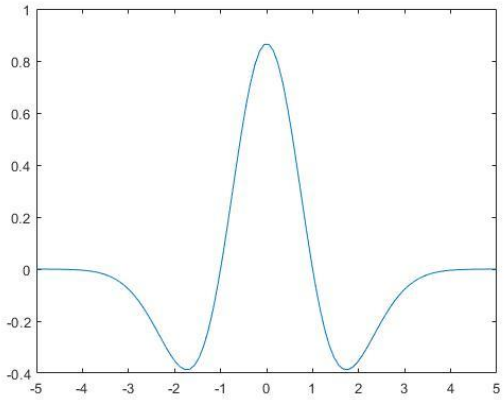

Figure 4.6: Mexican Hat wavelet.

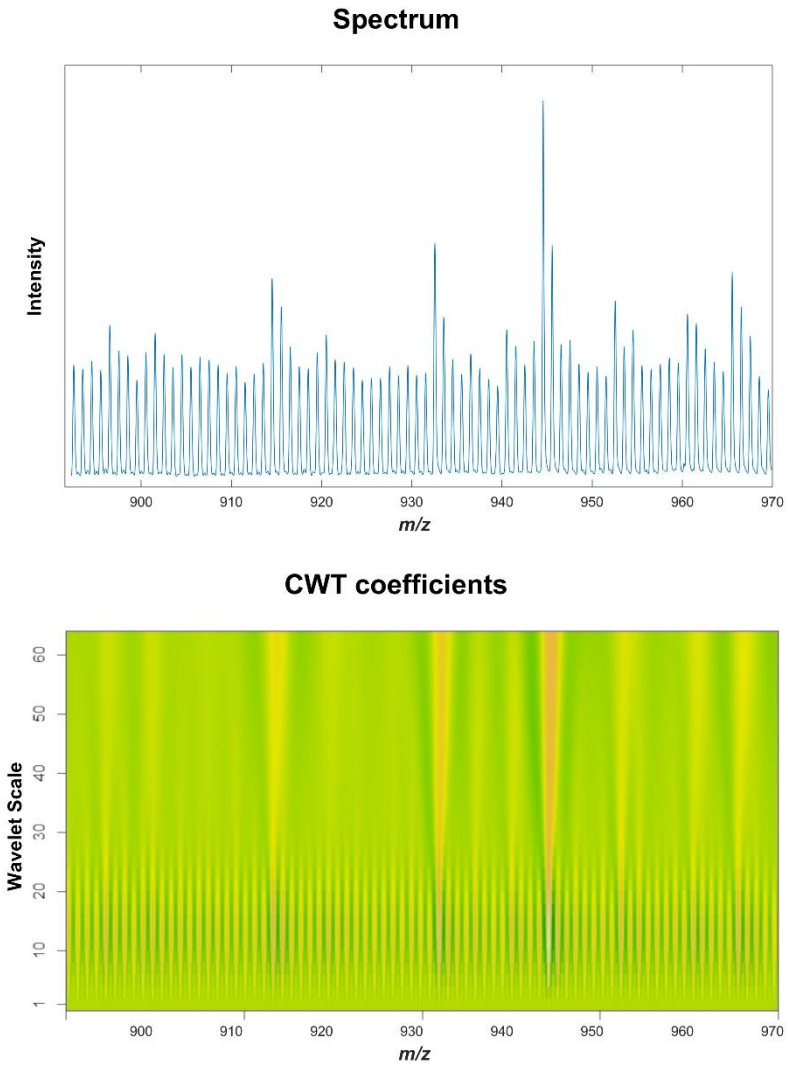

Figure 4.7: Heat map (bottom) of the coefficient matrix calculated for aggregated spectrum (top).

The advantage of using CWT instead of, a simple filter is that the shape of the wavelet changes with the scale. This is important because the shape of true peaks changes with the increase of *m/z*. At higher *m/z* values, peaks tend to be wider and smaller than at lower *m/z* values. Figure 4.7 shows how part of the spectrum looks in the wavelet space. For scales between 10 and 20 we see clear edges between peaks and valleys appearing in the signal. With higher scale values we can distinguish the groups of peaks that rise above the background.

The next step in this peak detection method is the identification of the ridge lines. It is done numerically by local maxima search. The search for maxima is done at each scale, with a moving window of width proportional to the o wavelet support region at the given scale. The local maxima are then connected to produce ridge lines. Details of the algorithm used to detect the ridge lines are in the paper by Du et al. [48]. Simply put, the algorithm starts the search from the largest scale (at the top), where each local maximum starts as a ridge line. The search continues by connecting ridge lines with local maxima from the next scale if they are within a range. If there is no continuation for a ridge line for a specified number of scales, it is removed. If a local maximum at a scale is not a continuation of any ridge line from previous scales, it is considered a new ridge line. After the ridge line identification is done, the peaks identified based on the length of ridge lines. Signal-to-noise ratio is calculated where signal is defined as the highest coefficient on the ridgeline within a range, and the noise is a 95^{th} quantile of the absolute CWT coefficient values at the lowest scale. Figure 4.8 shows the ridge lines for the coefficient matrix and the peaks ultimately identified with this method with the SNR threshold set to 3.

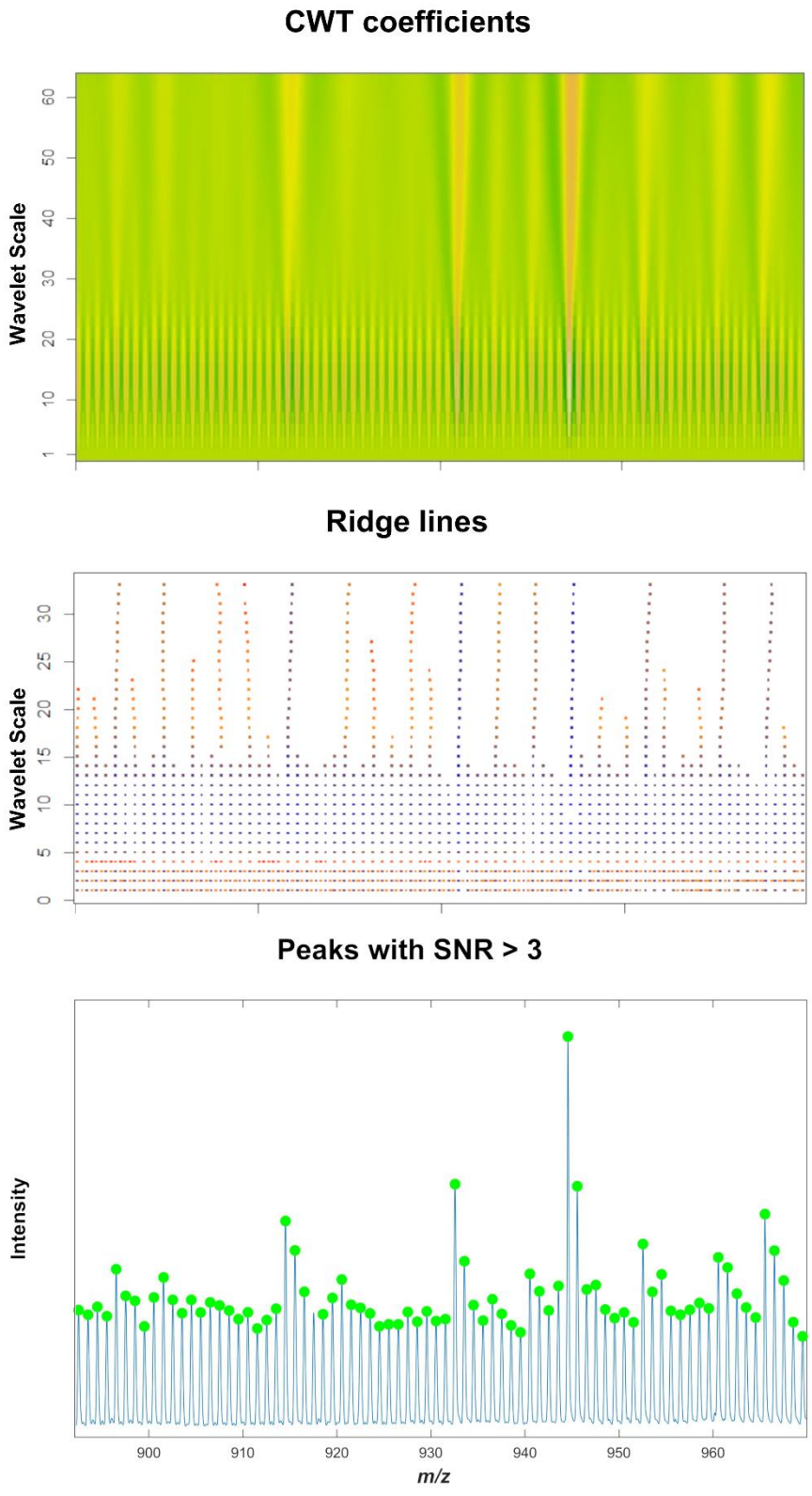

Figure 4.8: Coefficient matrix heat map (top), ridge lines (middle) and detected peaks (bottom) for a part of the signal.

5. Gaussian mixture model-based spectrum modeling

During the peak modeling method described in the previous chapter, the expected peak shape was used to transform mass spectra into wavelet space and then used for peak identification. Wavelets are well suited for modeling local features such as spectral peaks, but the coefficients and basis functions used to represent expected intensity have no inherent biological interpretation [62]. Gaussian mixture spectrum modeling [18, 63] aims to solve this problem. In this chapter, we describe the complete spectrum modeling process based on the GMM spectrum modeling method described in [62]. In this work, we propose different strategies to acquire the spectrum model than presented in the publication.

The idea of spectrum modeling is that each molecule in the mixture is represented by an element of the spectrum model. Of course, only in an ideal case each element of the model is correlated with a molecule in the mixture under study. In reality, most elements model only noise. Regardless of the source of the noise, whether it is from sample impurities, mass spectrometry imperfections, or some other source, these elements always exist. An important future of spectrum modeling is that each element of the spectrum model is described by a set of parameters that can be used to filter out noise and outliers. In the case of GMM-based spectrum modeling, the elements of the model are Gaussian distributions described by three parameters σ, λ and μ.

The fact that elements of the model can be easily interpreted as specific molecules, if the model element indeed is correlated with a molecule, is one of the greatest advantages of this approach. Another is the ability to filter the noise and select features not only based on the intensity of the peaks, as most peak detection methods do. This is why this method of peak detection is used in this work. Thanks to all this, the results of further analysis are easy to interpret and therefore can be useful in real applications. The main concepts involved in this chapter are wavelet-based peak detection, Gaussian mixture model decomposition, expectation maximization algorithm, structural analysis of images, and spatial distribution comparison.

5.1. Partitioning of the mass spectrum

The general idea of GMM-based spectrum modeling is described in [62]. In this publication, the Gaussian Mixture Model was fitted, using the Expectation Maximization algorithm, to the entire mass spectrum at once, creating a model in which each peak (model component) is described by a normal distribution.

This method of spectrum modeling can be used for mass spectra of simple mixtures containing a relatively small number of peaks. For more complicated mixtures, where the number of different molecules in the analyzed mixture reaches hundreds, it is better to first decompose the mass spectrum into smaller parts, as described for example in [18]. The final model of the spectrum is then the combination of the Gaussian mixtures fitted separately for each part. In [18], a method for splitting the mass spectrum into fragments is described. However, no clear goals or rules are defined for what the result of such a process should be. In this work, the process of splitting the signal into parts is studied in more detail. Rules for division are established, and various methods for finding such parts are tested.

5.1.1. Rules for partitioning of the mass spectrum

Due to the complexity of the samples being analyzed in this work and the resolution of MALDI-TOF mass spectrometry, the mass spectra potentially contain hundreds of true peaks that may overlap. For this reason, modeling of the spectrum using the Gaussian mixture model must be done independently for small portions of the signal and then combined into a general model. This raises the important issue of splitting the signal into these parts.

In the [18] a method for such a division is described. In this method the splitting of the mass spectrum is done with the help of 'splitters'. Splitters are the fragments of the mass spectrum defined as regions around 'clear' peaks, where 'clear' peaks are the peaks found by a peak detection method. In [18] 'mspeaks' function from the Matlab bioinformatics toolbox was used to find 'clear' peaks. Subsequently, the signal was split into parts using the splitters. This method successfully split the signal into smaller parts, however, it was not investigated whether the results of this method were suitable for the actual purpose of modeling the entire spectrum. To answer this question, the goal of the splitting method must be defined, and the criteria that determine a "good" part must be established.

The first property of a part of the mass spectrum that we want to study is its size. The reason for splitting the mass spectrum is that its size is too large for the EM algorithm to accomplish an optimal decomposition into the Gaussian mixture model. Ideally, a part would contain a single peak modeled by a single normal distribution. In this case there would be no randomness, due to the indeterministic nature of the expectation maximization algorithm, since only one normal distribution, that optimizes the likelihood, exists for that part. The more elements of the GMM there are, the stronger the effect of the random nature of the EM algorithm. Therefore, the smaller the part, the better.

However, as mentioned earlier, peaks describing different molecules may overlap, and such elements should not be split into different parts. In view of this, the parts should be as small as possible but a peak or series of overlapping peaks should be contained in a single part.

Figure 5.1 shows the averaged spectrum with a small section for which the optimal points of division were selected manually and the Gaussian mixture model of the part with overlapping peaks. What we can see is that, first, each peak that is most likely related to high-frequency noise is put in a single part. In the middle, a section that may contain a true peak or multiple peaks that overlap each other is put in the same part. In the last section (D) the result of the GMM decomposition of that part is shown. The gray lines show the individual components of the GMM model. It is clear that in addition to the elements that are correlated with noise, there are also elements that potentially model true peaks in the spectrum. The red line shows the sum of all GMM components.

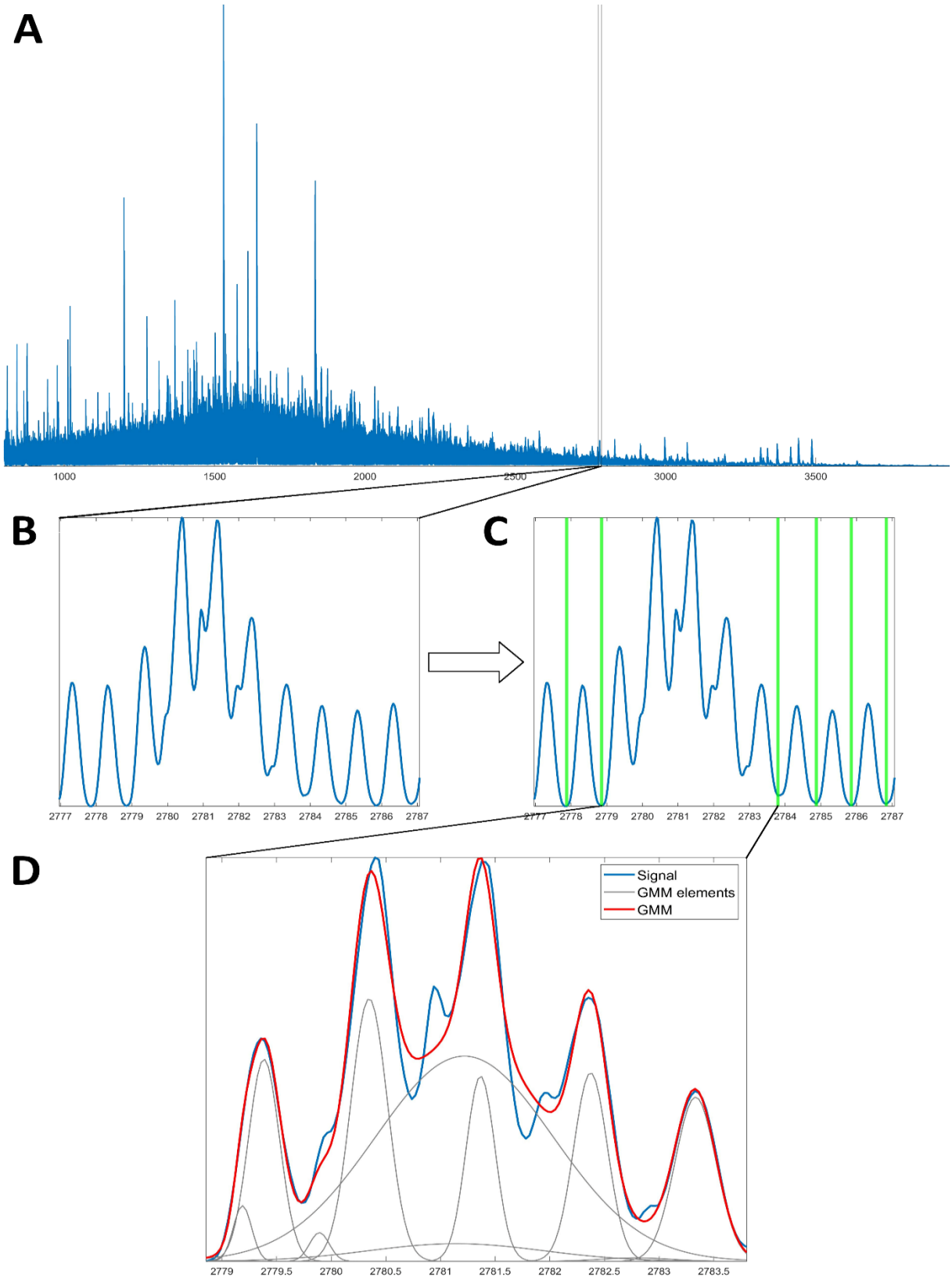

Figure 5.1: Example of the division of the mass spectrum and decomposition of the part with GMM. A - Entire mass spectrum. B - zoom into a mass spectrum. C – Example of good points of division. D – GMM of the single part of the spectrum.

5.1.2. CWT-based peak identification

Having defined the characteristics of a "good" part, the next step is to choose an algorithm capable of finding division points that split the mass spectrum into parts that meet these expectations. A logical solution to find such points is to look for peaks in the mass spectrum and use these peaks to divide the signal. This is the main idea of the "splitters" used in [18]. We opted to try a similar but simpler approach, where parts are created by finding peaks and then cutting the spectrum at the lowest point between adjacent peaks. In this case, we used the continuous wavelet transform-based peak detection method described in detail in the previous chapter. As it was described, in this method, peaks are detected by transforming the signal into wavelet space and then identifying ridge lines that indicate the positions of the peaks. The MassSpecWavelet package from the Bioconductor project was used to calculate the location of the peaks. After finding peaks, division points are identified by searching for the minimum value between neighboring peaks.

The results of this process are shown in Figure 5.2. The red stars mark the peaks discovered by CWT. In sections B and C, we zoom in on the smaller fragments of the mass spectrum for low and high *m/z* values. The green dots mark the locations where this method decided to cut the signal. It can be clearly seen that at high *m/z* values the gaps between the peaks increase. This is due to the fact that there are fewer true peaks in this part of the mass spectrum and the signal-to-noise ratio is lower at high *m/z* values.

The partitioning of mass spectra with this method is not ideal. Even though at low and medium *m/z* values the method succeeds in splitting the signal into small parts, at high *m/z* values, where there are few true peaks, it does not meet the first requirement for the "good" part. The fact that very broad parts are produced in regions of mass spectra where there are few true peaks is not a major problem, since most of these spectrum model elements are filtered in later phases. A far greater problem with this method is its inability to satisfy the second requirement for a "good" part. As can be seen in Figure 5.2 the regions containing potentially overlapping peaks are split into multiple parts. This becomes even clearer in Figure 5.3. Figure 5.3 A shows a previously selected small fragment of our mass spectrum with manually marked, ideally placed division points that put a single group of peaks into a single part. Figure 5.3 shows that the CWT-based method does not achieve the same results.

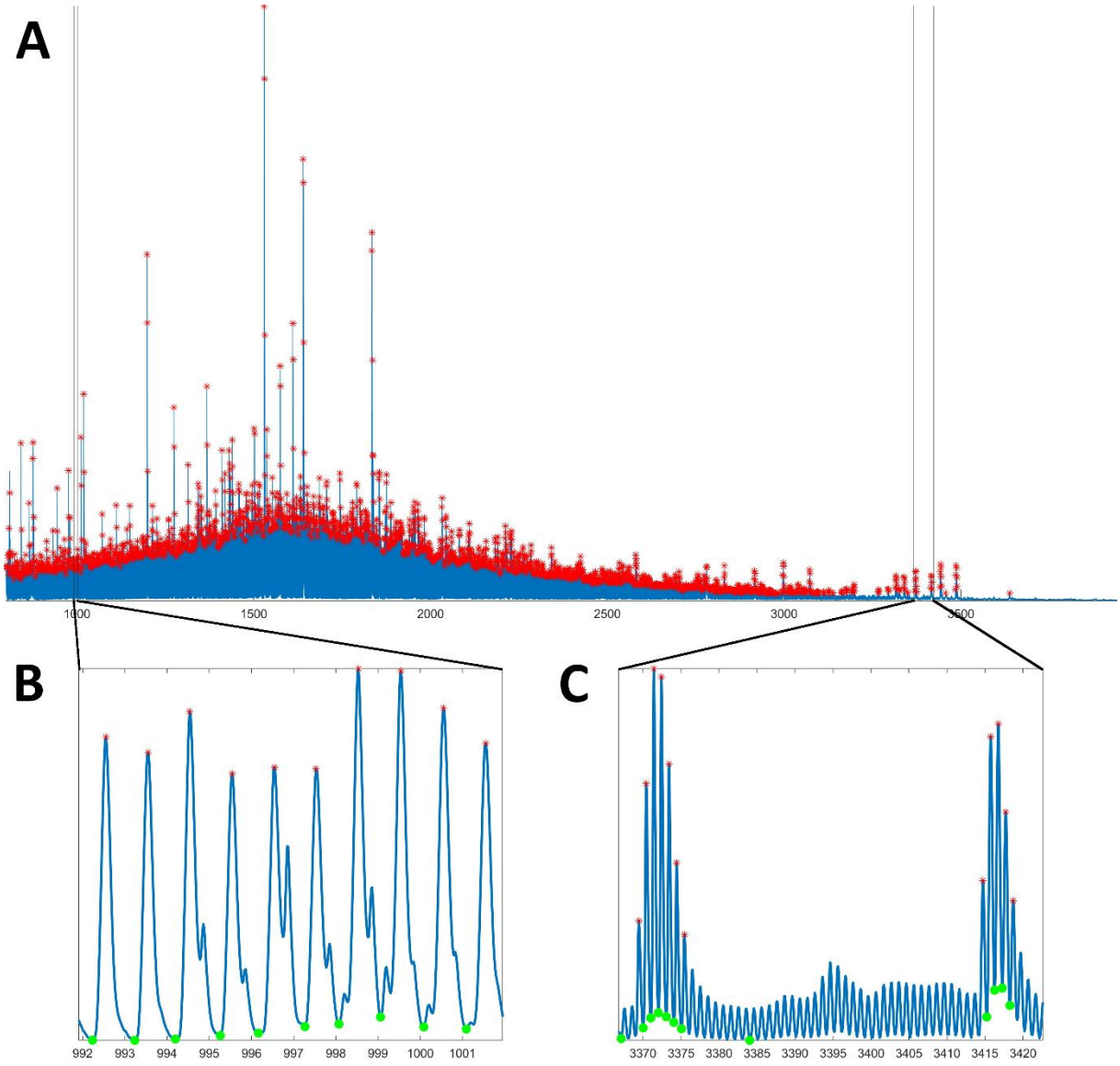

Figure 5.2: Identification of division points using CWT peak detection. A – peaks found for the entire mass spectrum, B – points of division identified in the section of the mass spectrum with low *m/z* values, and C - points of division identified in the section of the mass spectrum with high *m/z* values.

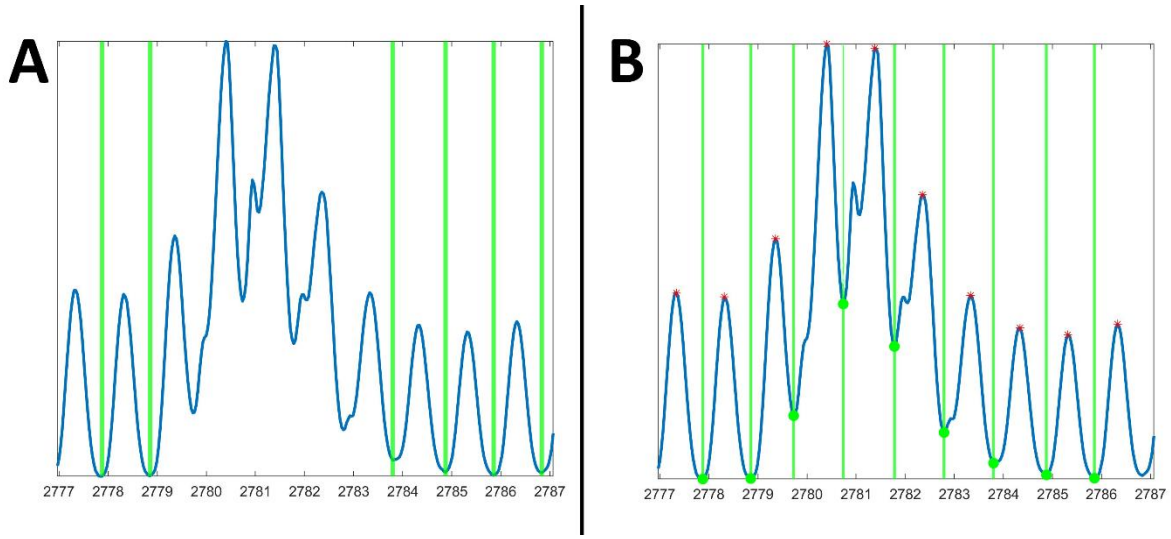

Figure 5.3: Comparison of the manual (A) and the CWT peak detection-based (B) division into parts of an exemplary part of the mass spectrum.

5.1.3. Identification of points of division

The approach for locating points of division by first finding peaks and then using them to locate points of division gave poor results. To find a better solution, a custom algorithm was developed that searches directly for division points without first searching for peaks in the spectrum. The algorithm attempts to find division points in a way that satisfies both requirements for "good" parts by performing a search for local minima with an additional constraint.

The algorithm searches for local minima based on the monotonicity of the signal and then selects division points at locations where the local minimum satisfies the constraint.

The constraint is set for the maximum value of the local minimum. To avoid splitting overlapping peak into separate parts, the value of the local minimum must not exceed a certain threshold. If the value of a local minimum exceeds the threshold, the local minimum is not considered as a split point.

The threshold is calculated for a local neighborhood and is equal to the 15^{th} percentile of all values in that neighborhood. As mentioned earlier, the resolution of the mass spectra changes with the increase of the *m/z* value. At the lower *m/z* values, the difference in atomic mass between neighboring channels is 0.018 DA and gradually increases to 0.041 DA at the end of the spectrum. For this reason, the width of the window for searching local minima changes and is expressed in a number of channels and not in DA. This means that the width of the window, measured in

Daltons, changes as the search progresses. The pseudocode for this algorithm is shown in Box 1.

```
BOX1 - Algorithm for the search of points of division
Output: POD – vector of indexes of points of division
Input: MS – mass spectrum (vector of intensities), W – threshold
window width, MinDist – minimum distance between points of division

BEGIN
  POD := [];
  isBT := false           // is the current value below
                          // the threshold
  idx := -1;
  FOREACH MSi DO           // for each mass channel
    isD := (MSi < MSi-1); // is the signal decreasing
    t:=calculate-threshold(MS,W,i);
                          // calculate threshold based on
                          // the local neighborhood
    IF (MSi<t AND !isBT)
      idx := i;      // remember the last index for
                     // which the signal dropped
                     // below the threshold;
      isBT := true;
    END IF
    IF (MSi>t AND isBT)  // if the signal increased above
                          // the threshold, find a new point
                          // of division
      isBT := false;
      window := MS(idx:i);
      PODIdx := min(window);
      lastPOD := POD(end);    // remember the last
                              // entry in POD vector
      IF (lastPOD + MinDist > PODIdx)
        IF (POD(end) > PODIdx)
          POD(end) := PODIdx;
                          // if the minimum found within
                          // MinDist is lower than the last
                          // entry in POD, then the entry is
                          // replaced
```

```
        END IF
      ELSE
        POD := [POD PODIdx];
      END IF
    END IF
  END FOREACH
END
```

```
Procedure – calculate-threshold
Output: t – threshold value
Input: MS – mass spectrum (vector of intensities), W – window width,
i – current index;

BEGIN
  IF (i <= W/2)                      // first W/2 elements
    window := MS(1:W);
  ELSE
    IF (i + W >= size(MS))
      window := MS(end-W:end);       // last W elements
    ELSE
      window := MS(i-W/2:i+W/2);
    END IF
  END IF
  t := prctile(window, 15);    // threshold value is
                               // the 15th percentile
                               // of the window
END
```

The result of this algorithm is shown in Figure 5.4. The green dots in section A mark all the division points made for the spectrum. The first thing to notice is that there are no large gaps as in the previous method. This means that the first rule for "good" parts is satisfied. Sections B and C are the same zoomed in fragments shown for the previous method. The most evident difference between the algorithms is visible in the section C with high *m/z* values. Single peaks are placed in separate parts, while the groups of peaks are put into a single parts.

In Figure 5.5 comparison between all three methods is made. Section A shows the manual points of division, section B the points found after CWT peak picking, and section C shows the points found by local minima search. We can see how the different methods split the same section of the mass spectrum. For this fragment of the

mass spectrum, the local minima based method succeeded in dividing the mass spectrum into optimal parts, while CTW-based method failed.

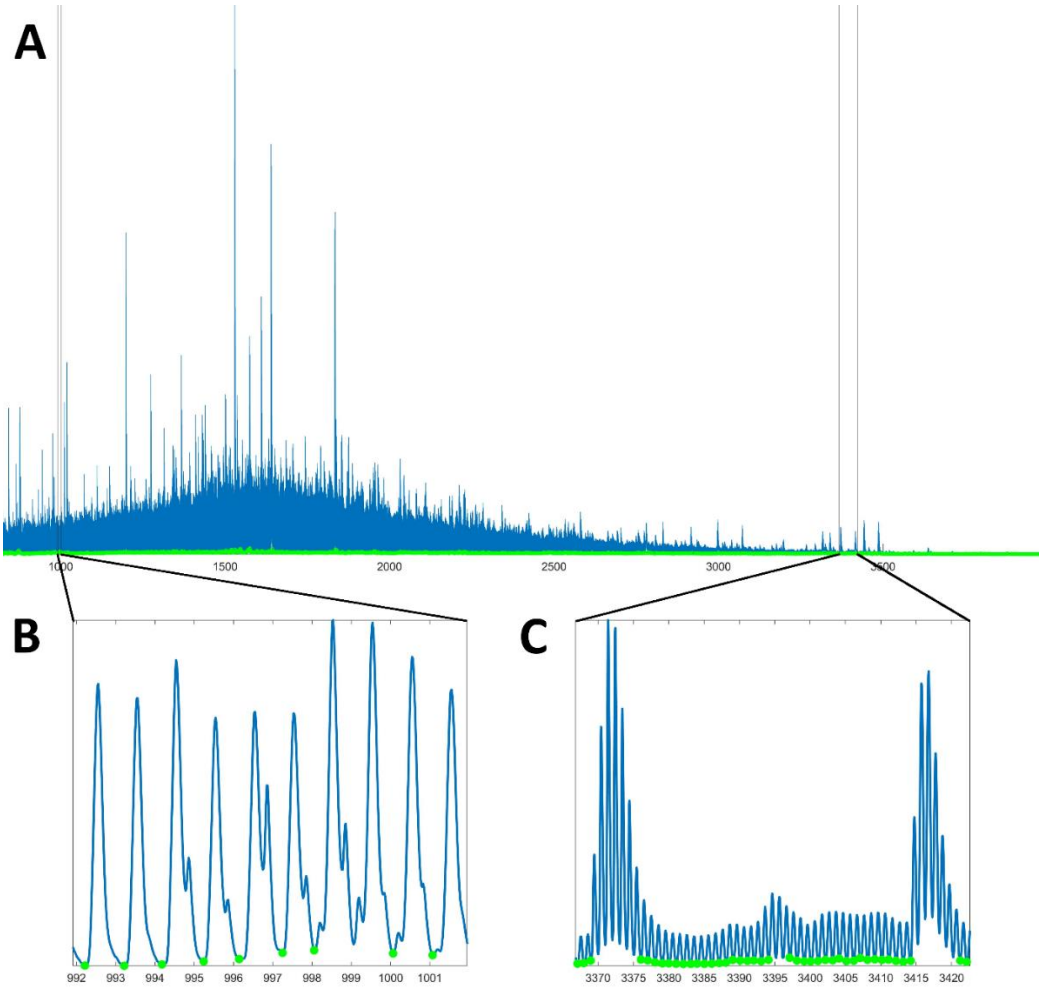

Figure 5.4: Identification of division points using moving window local minimum search. A – points of division identified for the entire mass spectrum, B – points of division identified in the section of the mass spectrum with low *m/z* values, and C - points of division identified in the section of the mass spectrum with high *m/z* values.

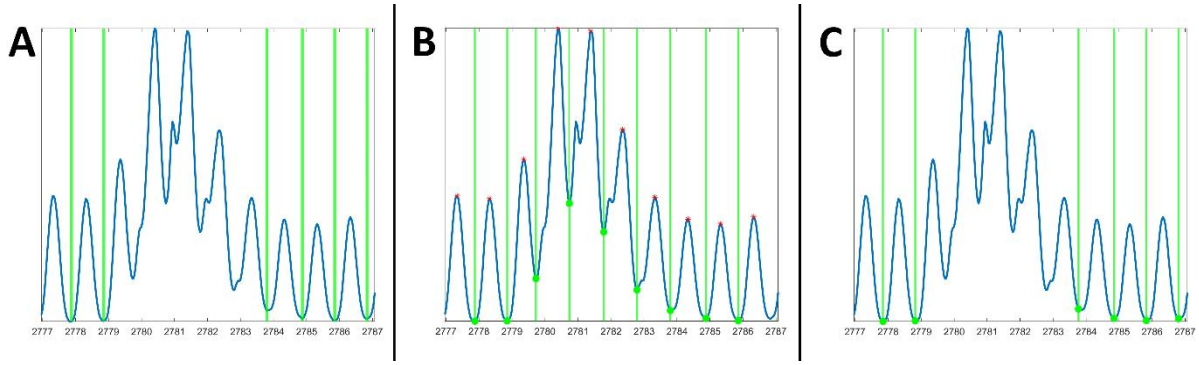

Figure 5.5: Comparison of the manual (A), CWT peak detection-based (B) and local minimum-based (C) division into parts of an exemplary part of the mass spectrum.

5.2. Gaussian mixture models

After splitting the signal, the next and most important step is fitting a Gaussian mixture model to each part. Each element in the GMM is described by three parameters: Mean (μ), Standard Deviation (σ) and Scale (λ). Mean and standard deviation describe the normal distribution of an element, and scaling provides information about the element share of the whole mixture model.

5.2.1. Estimation of the parameters with the EM algorithm

To find estimates of GMM parameters we maximize log-likelihood. defined by formula (4). Finding parameters that maximize log-likelihood is impossible analytically, therefore expectation maximization (EM) algorithm is used.

$$L(\Theta|x_1, \dots, x_n) = \sum_{i=1}^{n} \log\left(\sum_{k=1}^{M} \lambda_k n(x_i, \mu_k, \sigma_k)\right), \tag{4}$$

where Θ – model parameters (λ, μ, σ) , M – number of GMM elements, λ_k – weight of kth element, n – number of observations, x_i – ith observation, and $N(x, \mu, \sigma)$ is the normal distribution defined by the formula (5)

$$N(x, \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(x-\mu)^2}{2\sigma^2}}. \tag{5}$$

The EM algorithm is a well-established method for estimating unknown parameters. This algorithm follows an iterative approach to reach a local optimum. Initially, the parameters of the model can be chosen either randomly or by data-driven approximation. During iterations, the values change and gradually improve until they reach a stop condition.

EM algorithm finds parameters by alternating between expectation (E) and maximization (M) steps. The expectation step of EM algorithm calculates the probabilities that an observation belongs to each GMM element. The probability that an observation belongs to the k^{th} element is given by the equation 6.

$$P_{k,x} = \frac{\lambda_k n(x, \mu_k, \sigma_k)}{\sum_{k=1}^{M} \lambda_k n(x, \mu_k, \sigma_k)} \tag{6}$$

This is followed by the maximization step, where the new values of the model parameters are calculated using the following equations.

$$\hat{\mu}_k = \frac{1}{\sum_{i=1}^{n} P_{k,x_i}} * \sum_{i=1}^{n} P_{k,x_i} * x_i, \tag{7}$$

$$\hat{\sigma}_k^2 = \frac{1}{\sum_{i=1}^{n} P_{k,x_i}} * \sum_{i=1}^{n} P_{k,x_i} * (x_i - \mu_k)^2, \tag{8}$$

$$\hat{\lambda}_k = \frac{\sum_{i=1}^{n} P_{k,x_i}}{n}, \tag{9}$$

where $\hat{\mu}_k$ – new mean of k^{th} element, $\hat{\sigma}_k^2$ – new variance of k^{th} element, $\hat{\lambda}_k$ – new scale of k^{th} element.

EM algorithm ensures that the likelihood calculated with parameters of the next iteration is not worse than the previous iteration. If the likelihood doesn't improve in the next iteration, the algorithm reaches a local optimum, and the search stops. Another way used to terminate the EM loop is to set a threshold value for the improvement of the likelihood score. Figure 5.6 shows an example of how the algorithm works.

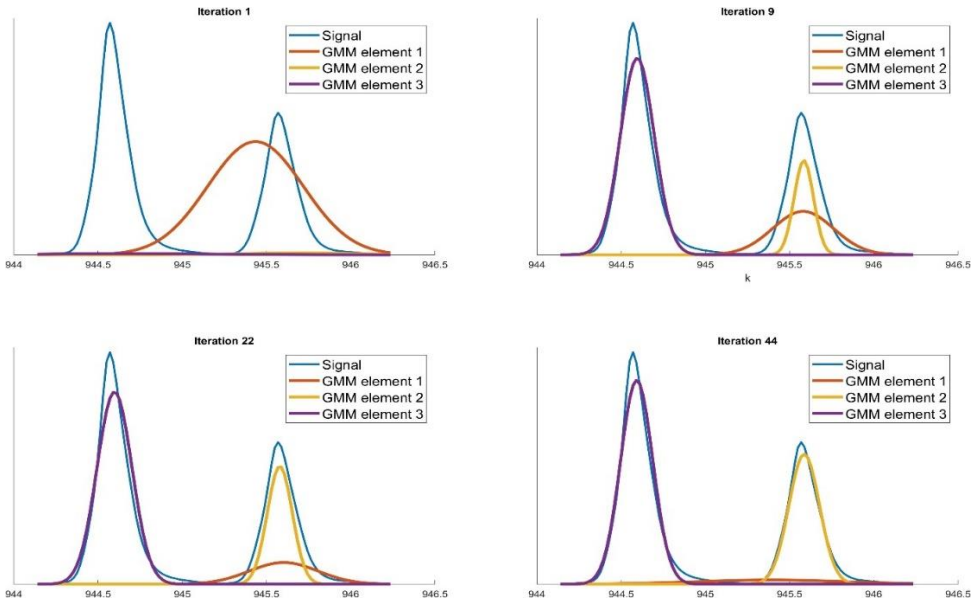

Figure 5.6: Gaussian mixture model during iterations of EM algorithm.

In this work, we used a custom implementation of the EM algorithm that works quickly with our particular type of input data, where the intensity in the mass spectrum describes the number of elements in the total population for a given *m/z* value. This means that the size of the population is very large, but the number of different values is several orders of magnitude smaller. The execution time of our implementation depends only on the number of distinct values in the population and not on the total size of the population. The implementation of the EM algorithm was written as a MATLAB function. A simplified pseudocode for this implementation is shown in Box 2.

```
BOX2 - EM Algorithm
Output: params – GMM parameters
Input: N – number of elements in GMM, values – 2D array of values
and their numerical strength (intensity)

BEGIN
  params := randomly choose initial GMM parameters;
  init_L := calculate log-likelihood for given values and
            params;
  ε = -Inf;
  shift = 0;
  WHILE (shift > ε) //EM algorithm loops until the shift
                    //between iterations is smaller than
                    //the value of ε.
    IF (any(params.sigma < 10^3))
      GO TO BEGIN
      //Rarely, the EM algorithm gets stuck in a loop where
      //σ of one of GMM elements goes to infinity, giving
      //useless results. In such a case, the algorithm
      //restarts.
    END IF
    params := calculate new GMM parameters;
    L := calculate log-likelihood using new params;
    shift := L – init_L; //Calculating the shift between
                         //iterations.
    IF (ε == -Inf)       //Initializing ε during the
                         //first iteration.
      ε := shift / 10^3;
```

```
    END IF
    init_L := L;
  END WHILE
END
```

5.2.2. Choosing the number of GMM components

One of the inputs of the EM algorithm is the parameter *N*. This parameter defines the number of Gaussian elements that the algorithm tries to fit into the data. Finding the optimal value for the parameter *N* is crucial. To find out how many elements should be included in the GMM of a given part, a few things must be considered. In general, we try to choose a number of elements that results in a model that describes the data as well as possible while being as simple as possible to avoid overfitting. The approach is to add elements to the GMM and examine how the likelihood of models changes with more and more elements.

Additional problem is that the EM algorithm is an indeterministic algorithm, i.e., the likelihood of models with the same number of elements changes each time the algorithm is run. To determine the extent to which the results change with each run and the effect this has on the likelihood, four random parts of the signal were selected for investigation. For each part, a GMM was fitted with up to ten elements. For each number of elements, the EM algorithm was repeated 1000 times. Figure 5.7 shows the boxplot of the log-likelihood values for this experiment. The plot shows that the variance of the log-likelihood is large for a small number of elements and then gradually decreases. The average log-likelihood increases with *N*, but the rate of improvement decreases.

The second problem is that adding an element to a GMM usually leads to a better likelihood score because more complicated models can be better fitted to the data. This is a pervasive problem in data modeling called overfitting. The solution to this problem is to introduce a penalty for model complexity into the assessment of the goodness of fit of the model. A commonly used value that introduces a penalty for model complexity is the Bayesian Information Criterion (BIC) (equation 10).

$$BIC = k * ln(n) - 2\,ln(L), \tag{10}$$

where k is the number of parameters in the model, L is the likelihood of the model and n is the number of observations.

The number of parameters in the model (k) depends on the number of elements in the model and the number of parameters of each element. As mentioned earlier, each element of the GMM is described by three parameters: mean (μ), standard deviation (σ), and scale (λ). Considering that the sum of all λ-values is 1, we calculate the k with equation 11.

$$k = 3N - 1 \tag{11}$$

where N is the number of elements in the GMM.

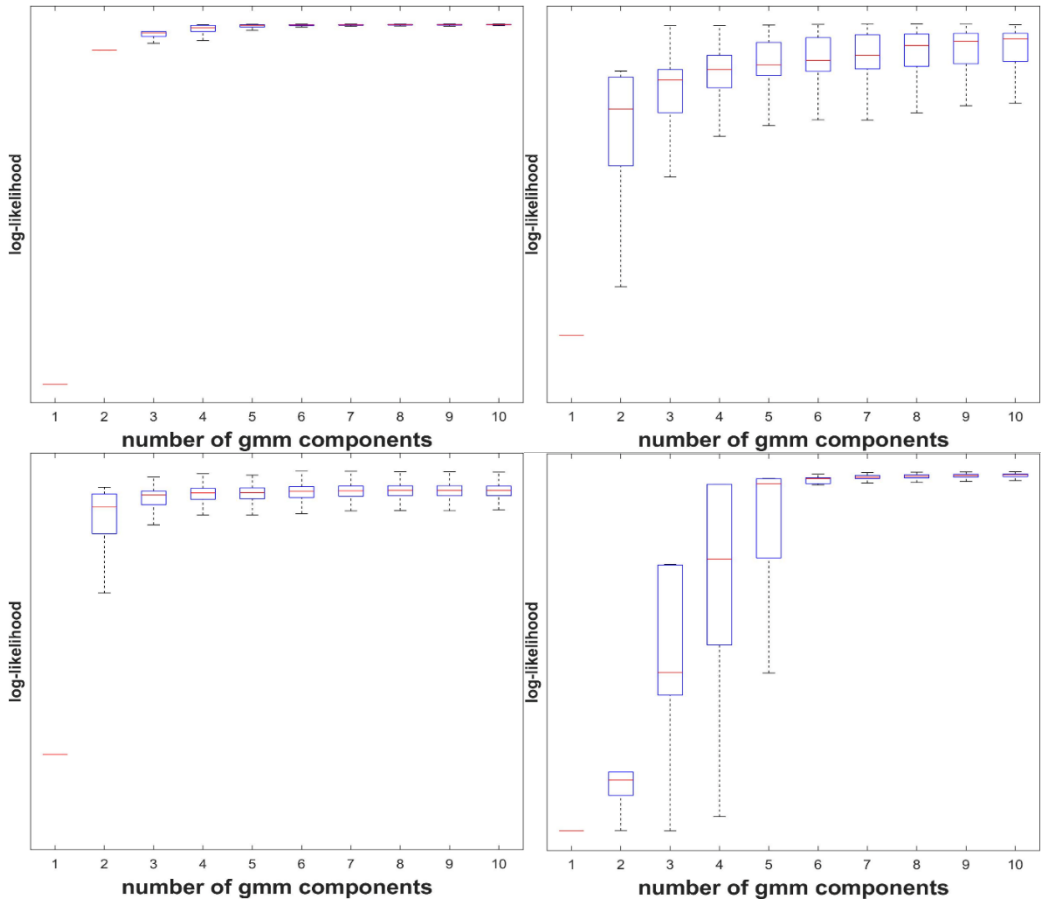

Figure 5.7: Graphical demonstration of the GMM complexity on the log-likelihood of the models for four random parts of the spectrum model. Each box plot represents the results for a single part fitted with Gaussian mixture models with up to ten elements.

After considering these problems, we decided that the final number of GMM elements for a part is chosen as follows. The EM algorithm is used to fit the part with a GMM, then the number of elements in the model is increased and the BIC value for the new model is calculated. If the BIC value of the model with an increased number of elements is worse than that of the previous model, the search is completed and the N of the previous model is the result. Due to the stochastic nature of this process, it is repeated several times. From the distribution of the entire population of results the

final *N* parameter is chosen as the one appearing most often. Figure 5.8 illustrates this process. Section A shows the analyzed fragment of the mass spectrum, section B plots the average BIC for each number of model elements and section C shows the distribution of the results. Ultimately chosen value of *N* is marked with color red.

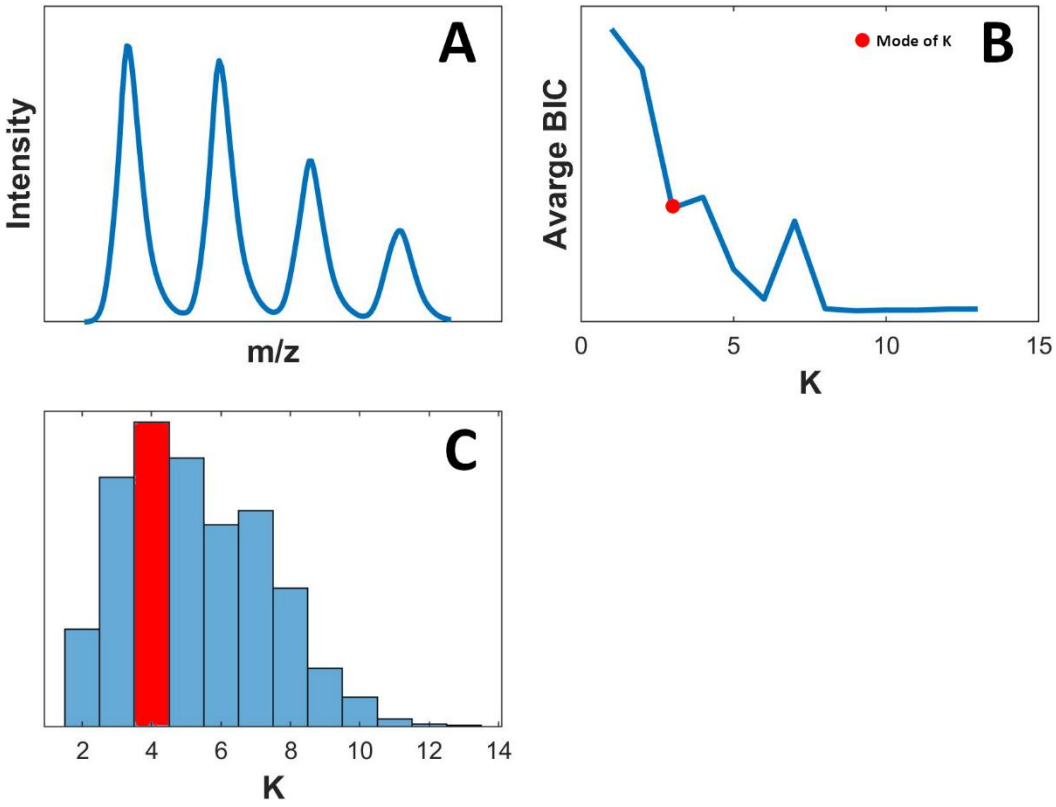

Figure 5.8: Results of the *N* search. Part of the signal (A), Plot of the average BIC (B), Histogram of *N* values (C).

5.2.3. Fitting Gaussian mixture models

Finally, when the value of parameter *N* is selected, the part is fitted with a GMM. This process is also repeated a few times to get the best result. After this is done for each part, we obtain the final spectrum model. The model (see Figure 5.9) consists of 9454 elements. The number of elements in the spectrum model changes very little when the entire algorithm is run again. The changes are, once more, the result of the indeterministic nature of the EM algorithm, but because of the steps taken during the modeling process, these changes are insignificant in the scale of the entire model. The changes in the number of elements of the spectrum model are around 1% of the size. The pseudocode of the entire spectrum modeling is presented in Box 3.

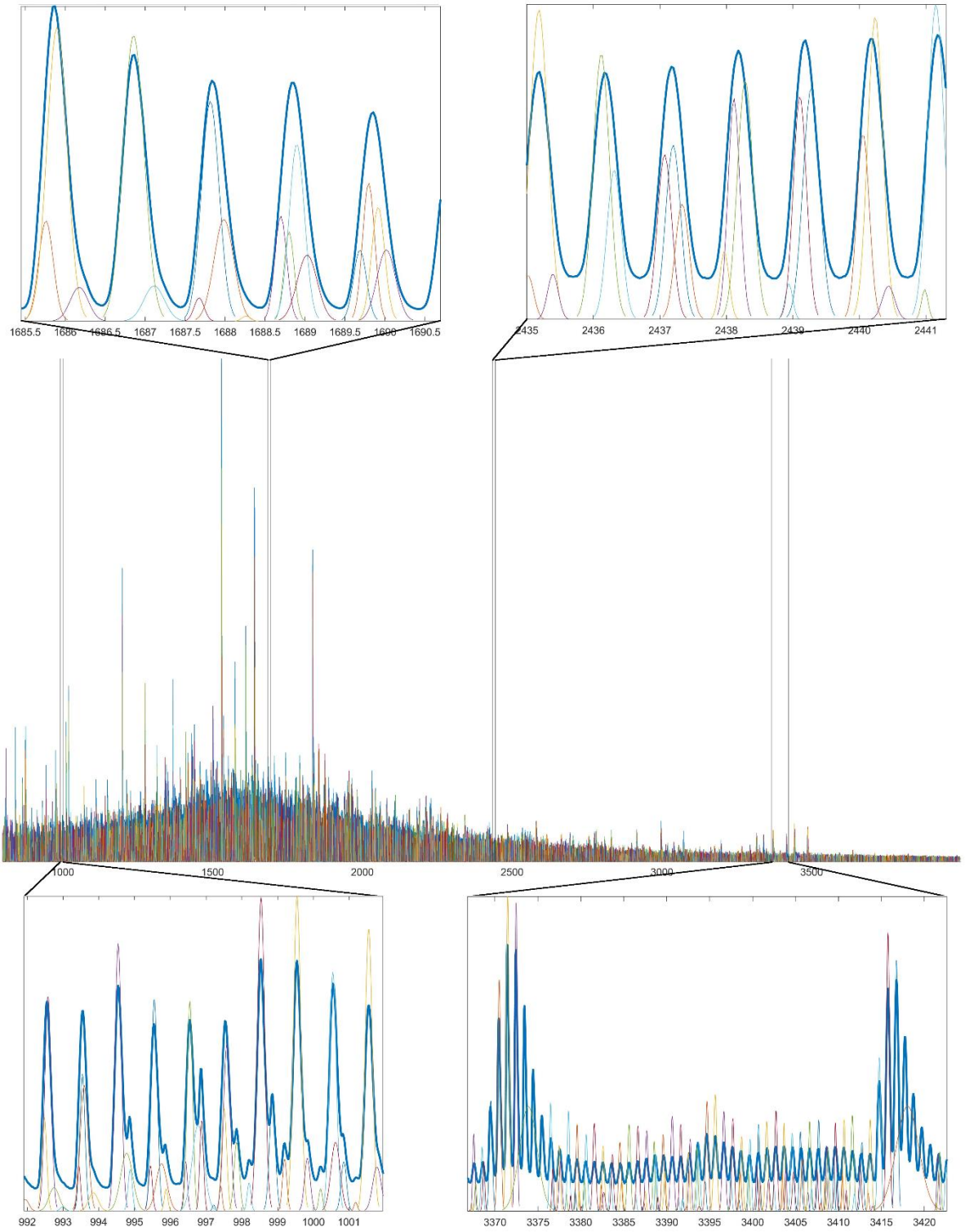

Figure 5.9: Final spectrum model.

```
BOX3 - GMM Modelling
Output: model – Array of model parameters
Input: POD – vector of indexes of points of division
BEGIN
  parts_array := divide_signal(POD);
  FOREACH part in parts_array DO
    part_likelihood := -inf;
    part_parameters := [];
    values := 2D array of part values and their numerical
              strength (intensity)
    k := find k value;
    FOR (100 times)
      gmm_parameters := EM_Algorithm(k,values);
      likelihood := calculate_likelihood(gmm_parameters);
      IF likelihood > part_likelihood
        part_likelihood := likelihood;
        part_parameters := gmm_parameters;
      END IF
    END FOR
    model := add part-parameters to array;
END
```

6. Feature engineering

The spectrum model is a set of normal distributions (components), and each distribution is described by the parameters σ, μ, and λ. Using these parameters, we can calculate the values of the features. Ideally, each component would represent a particular molecule in the mixture under study. However, a large fraction of the components are present at this point only because of the high frequency noise in the signal. The number of components in the spectrum model is 9454. This can be treated as a set of features and be used to train a classifier, but for most machine learning methods this number is still too high to train a classifier in a reasonable amount of time.

For this reason, we try to filter out the components correlated with the noise in the mass spectrum. At this point it is important to remember that filtering the noise also potentially removes features correlated with low intensity true peaks and some of these low intensity peaks may have a great impact on the predictor. For this reason, it may be beneficial to train classifiers on the unfiltered feature set if the time is not a great concern, especially because biomarkers correlated with low-intensity peaks may still be unknown to specialists. However, the time and computational power required for such methods might be too high for such an approach to be useful for a diagnostic test. In the next steps, we aim to further reduce the dimensionality of the data by filtering out some of the components.

6.1. Noise filtering

Analysis of MALDI-TOF MS data is a multistep process. When performing multiple experiments small differences (shifts) appear between the mass spectra. These differences can be caused by the calibration of the instruments or the handling of the samples and instruments [45]. This problem occurs mainly when there is a larger time interval between experiments. This problem is handled at the beginning of the data

processing with spectrum alignment. Another key problem with the data is the noise. The noise has many sources and each step of MS can add to the problem. Denoising can be done in many different ways. For example the noise is often filtered during other preprocessing steps such as baseline correction or peak detection.

In this work, the first step to remove the noise is to remove the Gaussians with values of λ below the noise level. The parameter λ is directly related to the intensity of the peak. Filtering based on the peak intensity is the basis of most, commonly used methods. We examined how the values of this parameter are distributed in the model. Figure 6.1 shows the distribution of λ values in the entire spectrum model. The highlighted threshold is used to filter low intensity components. We assume that the left part of the distribution models the elements that model the noise, and the right part is correlated with true peaks in the spectrum. As we can see, removing the noise components also removes some of the true peaks. Nevertheless, removing the first element of the distribution removes only some of true peaks and most of the noise from the signal. The number of elements in the model was reduced from 9560 to 2884. Figure 6.2 to Figure 6.4 show parts of the spectrum model before and after noise filtering.

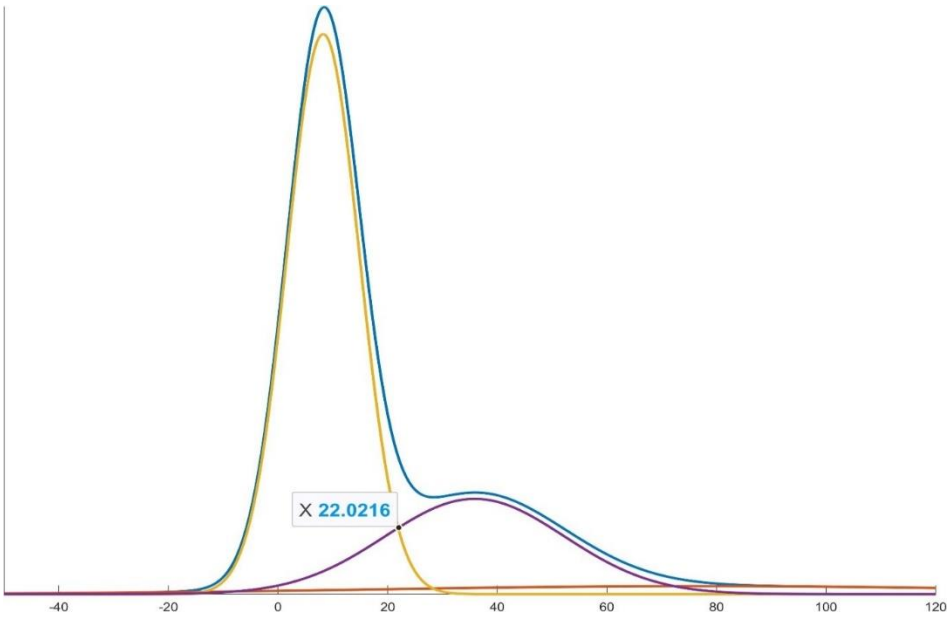

Figure 6.1: Distribution of λ values.

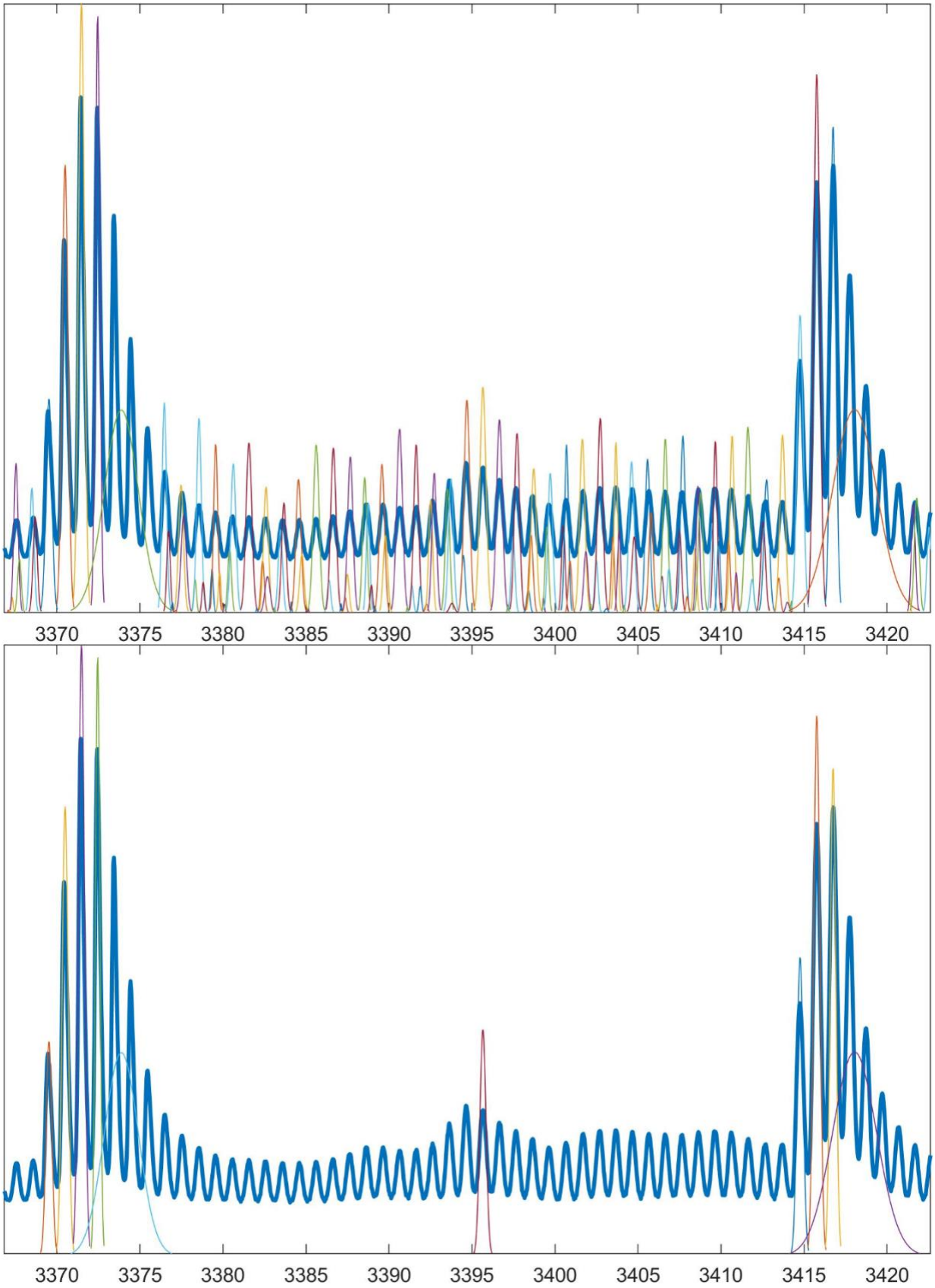

Figure 6.2: Aggregated mass spectrum with spectrum model before (top) and after (bottom) noise filtering. The displayed part shows a fragment of the spectrum with high *m/z* values.

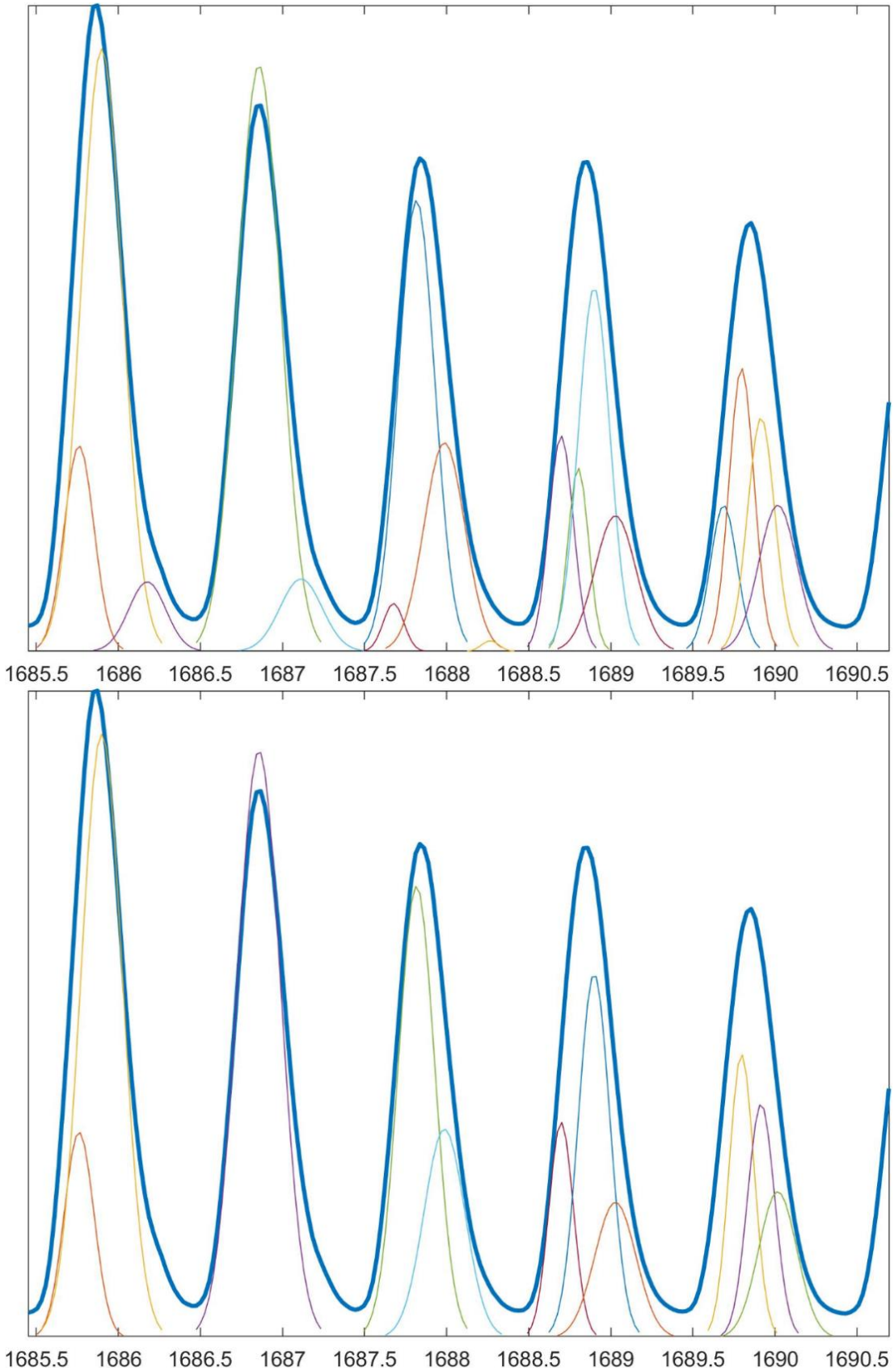

Figure 6.3: Aggregated mass spectrum with spectrum model before (top) and after (bottom) noise filtering. The displayed part shows a fragment of the spectrum with medium *m/z* values.

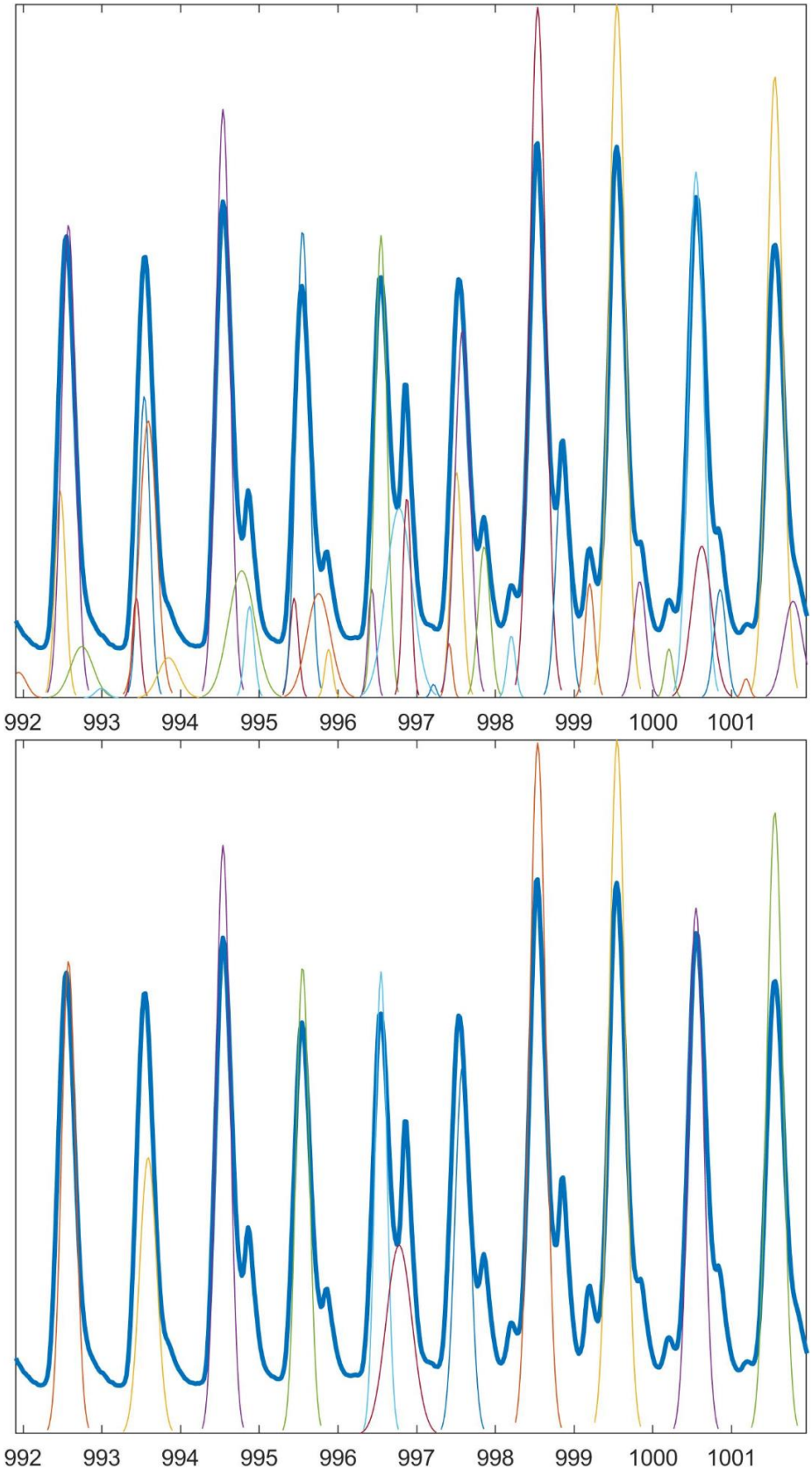

Figure 6.4: Aggregated mass spectrum with spectrum model before (top) and after (bottom) noise filtering. The displayed part shows a fragment of the spectrum with low *m/z* values.

6.2. Feature selection con.

The number of elements (features) after filtering has decreased substantially, but is still far from the expected number of several hundred. At this point, a crucial part of the proposed workflow begins, namely feature engineering based on the spatial distribution of features.

This feature selection phase is a two-step process. The first step compares the spatial distribution of nearby features. Figure 6.5 shows a simple part of the aggregated spectrum. As we can see, it contains three components, but the part looks like a single peak and perhaps should be described by a single normal distribution. There may be several reasons why this part was split into three components when modeling with the Gaussian mixture. One is that after aggregation of the mass spectra, imperfections in the baseline correction are more significant and the offset has led to this situation. Another is the fact, that the shape of the peak is not an ideal Gaussian distribution. Due to uncertainties in peak detection caused by slight variations during ion motion in the magnetic field, the true peaks in the mass spectrum are slightly skewed. This is characterized by a slight flattening of the right slope of the peak. This can result in a single peak being broken down into multiple components. Finally, it may be that the part actually describes two or more overlapping peaks and it should be described by two or more components in the spectrum model.

To decide whether nearby components are correlated with a single molecule, we compare how these components are spatially distributed among the samples. We assume that if there is no statistical difference between the spatial distributions, these components are correlated with a single molecule of the mixture. In such a case, the components are combined into a single feature. The component with the smaller area is removed and the larger one remains with the new area (value of the feature) as the sum of the areas of the two components. If, on the other hand, there is a statistical difference in the spatial distribution of the components, then we assume that they model different molecules, and both remain in the spectral model.

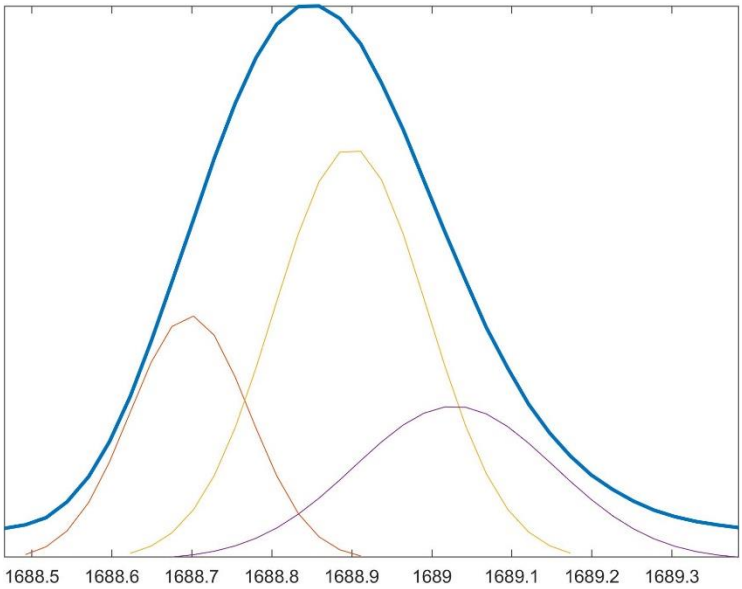

Figure 6.5: Potentially a single peak decomposed into multiple GMM components.

6.2.1. Peacock's test

To find out whether a population has a certain distribution, we use a statistical test. Such a test checks whether there is a significant difference between distributions by calculating a value, called the test statistic, and comparing it to a particular distribution, such as for example, the chi-square distribution, and then accepting or rejecting the null hypothesis at the chosen significance level. This is the best practice when comparing an empirical distribution with a theoretical distribution, for example, when checking whether the empirical data is normally distributed. There are many different tests that can be used depending on the specifics of the task at hand. To compare two empirical distributions, a good choice is the Kolmogorov-Smirnov test [64].

The Kolmogorov-Smirnov (K-S) test can quantify the distance between two empirical distribution functions of two samples. Figure 6.6 shows the visualization of the K-S test statistic. The figure shows two artificially generated distribution functions. In the example, the value of the K-S test statistic is 0.28. Using the tables, the *p*-value is 0.0317 at 5% significance level. This means that we reject the null hypothesis at the 5% significance level and conclude that the samples are not similarly distributed.

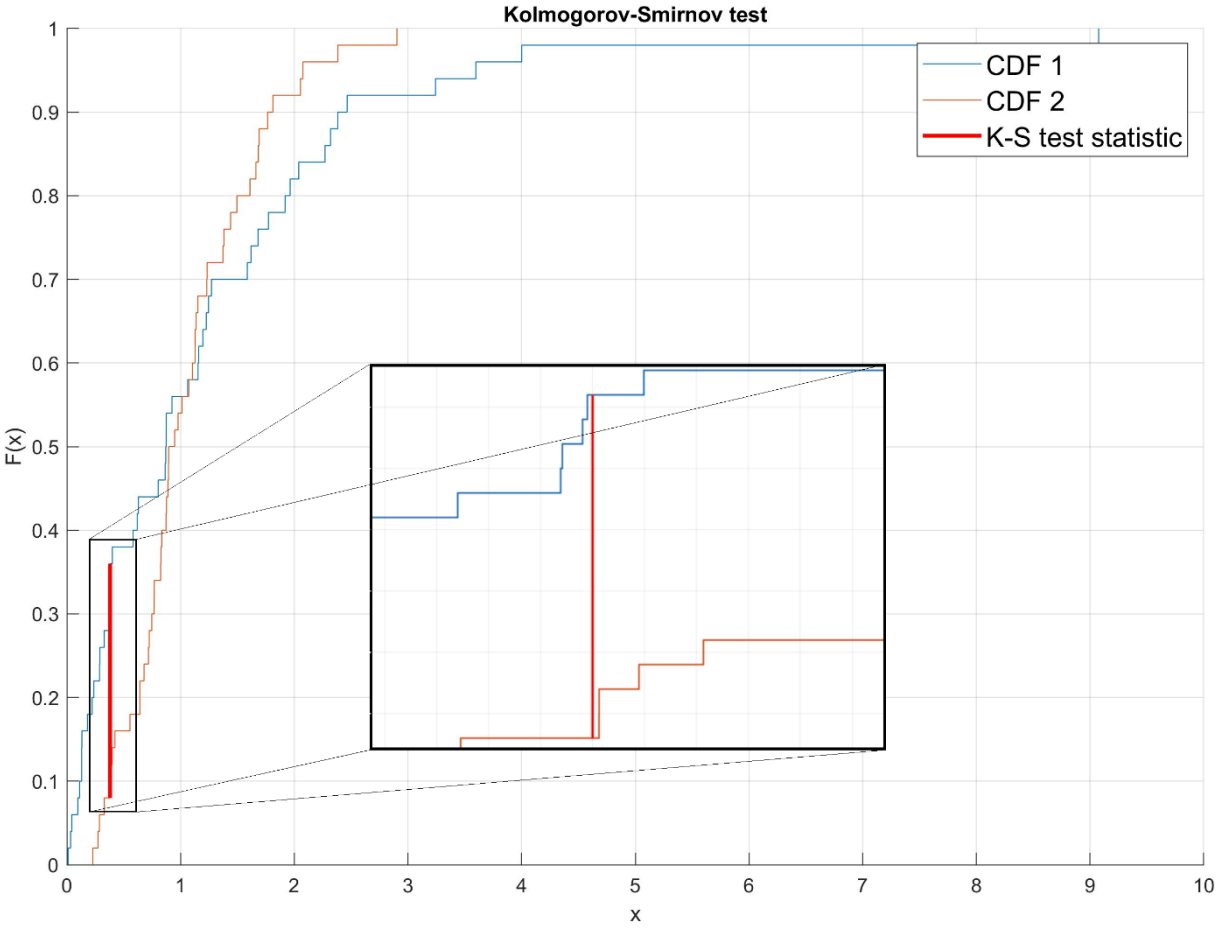

Figure 6.6: Kolmogorov-Smirnov test example.

For comparing the distribution of our components, the Kolmogorov-Smirnov test is inadequate because it can only compare one-dimensional data, whereas we are trying to compare two-dimensional spatial distributions. In his paper Peacock at el. [65] described a method for comparing two-dimensional distributions with the extension of the Kolmogorov-Smirnov test. We use Peacock's test to decide whether the spatial distributions of two components are statistically different.

To better illustrate how the Peacock test statistic is calculated, we compared two random components and showed the steps to calculate the value of the Peacock's test statistic. Figure 6.7 shows the spatial distribution of the randomly selected components numbered 16 and 931. It is clear by visual inspection, that their spatial distribution on the sample is completely different.

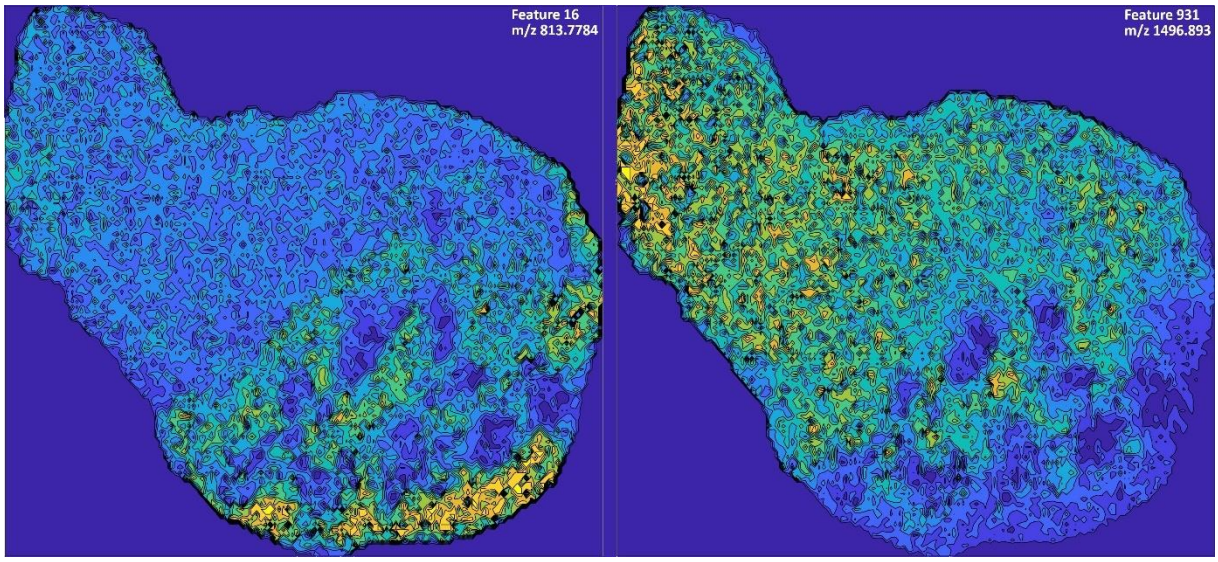

Figure 6.7: Spatial distribution of component number 16 and 931 on sample 1.

The method by which the value of the Peacock's test statistic is calculated is as follows. For each pixel of the image, the difference between the continuous distribution functions in four directions is calculated (see Figure 6.8). Each section of the figure illustrates the difference between the continuous distribution functions in each direction. The highlighted pixel is the location where the difference is the highest in all four directions. Figure 6.9 shows the difference in all directions at once. The value of each pixel is the maximum difference for that pixel in any direction.

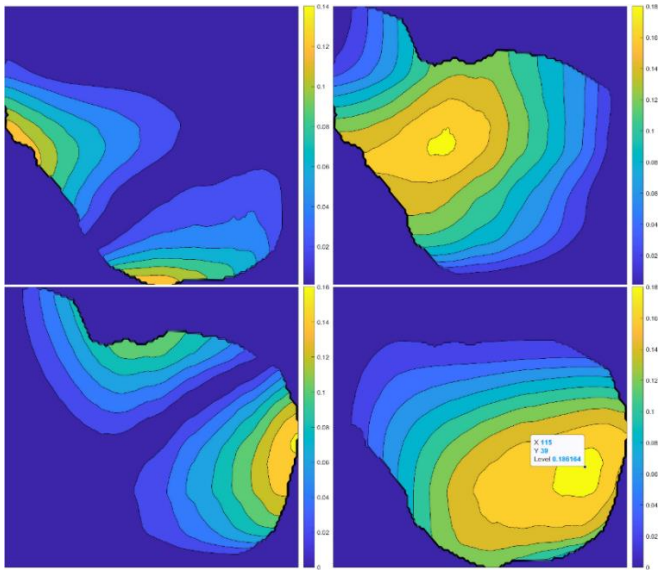

Figure 6.8: Difference between features CDF in each direction.

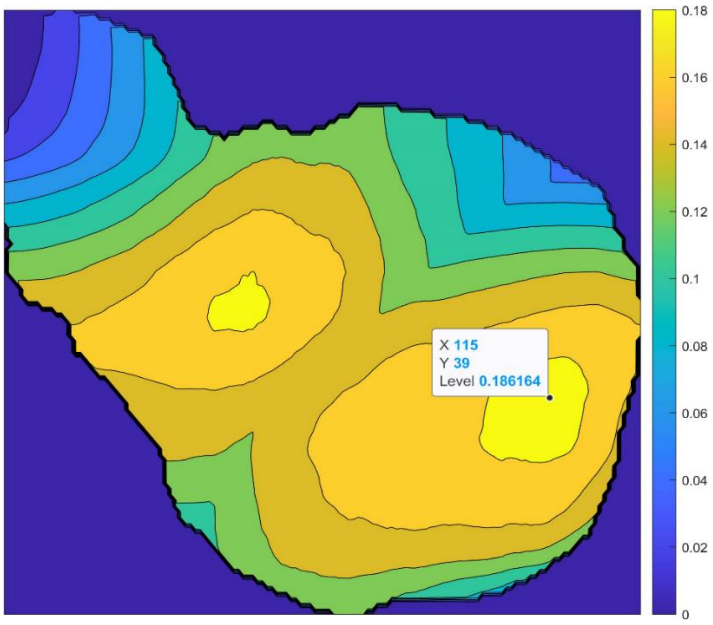

Figure 6.9: Overall difference in compared features CDF's.

A critical value can be calculated for the K-S test, but significance levels cannot be established for the Peacock's test, especially for very large sample size [22]. To find the critical values, we performed numerical simulations. For each sample, we performed a permutation test to find the empirical distribution of the Peacock's test statistic. We computed several thousand values of the Peacock's test by comparing random pairs of features. Figure 6.10 to Figure 6.13 show how these values are distributed for each sample.

As we can see, the distributions for each sample are very similar. The critical values are different, of course, because each sample has a different size. The distribution of Peacock's test statistic in each sample consists of three components, representing three categories of Peacock's test scores. We assume that the left component models tests for very similar spatial distributions. The middle component models reasonably similar spatial distributions, and the right component models the Peacock's test statistic values for completely different spatial distributions.

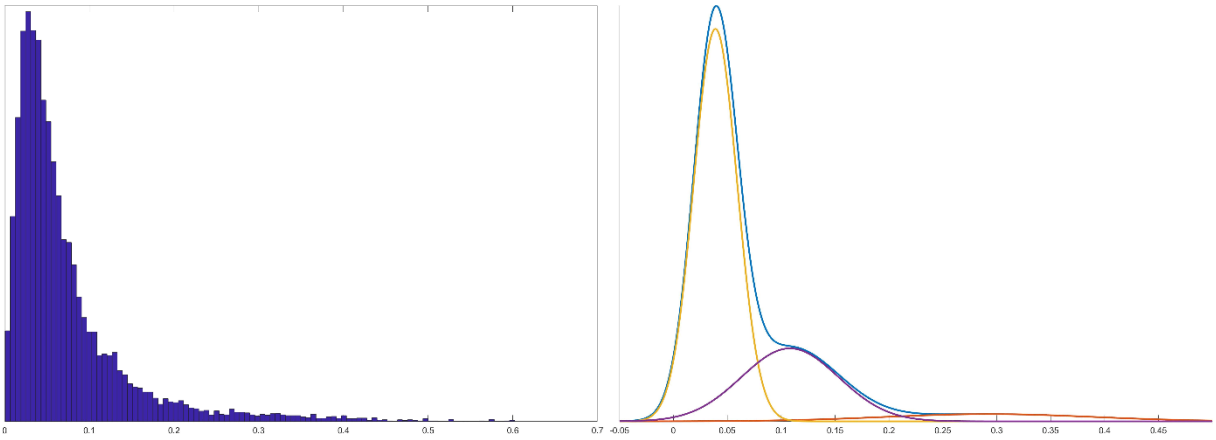

Figure 6.10: Empirical distribution of Peacock's test statistic for sample 1.

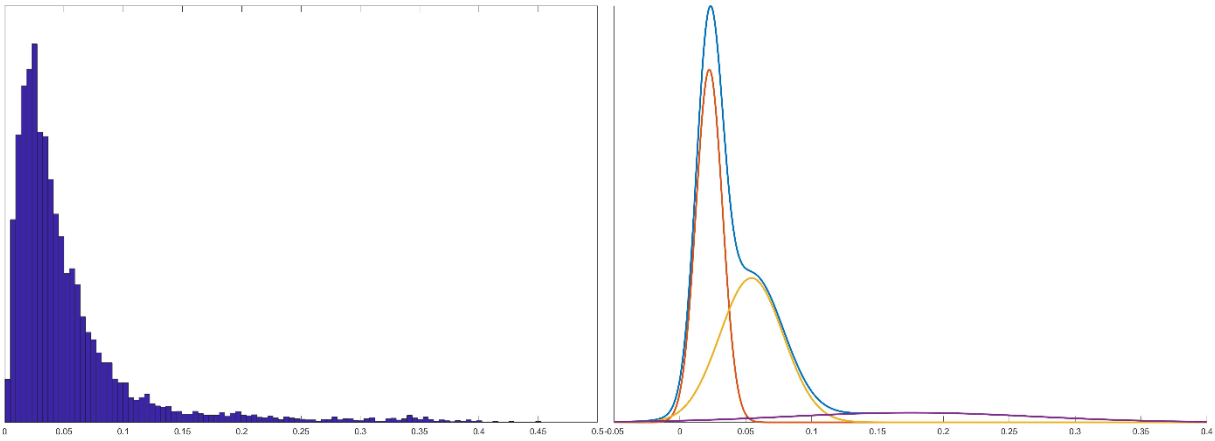

Figure 6.11: Empirical distribution of Peacock's test statistic for sample 2.

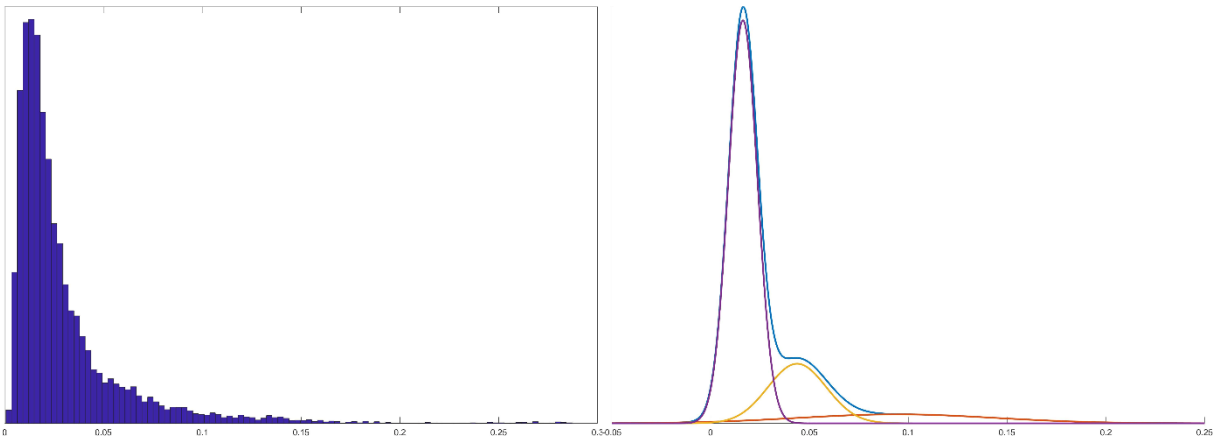

Figure 6.12: Empirical distribution of Peacock's test statistic for sample 3.

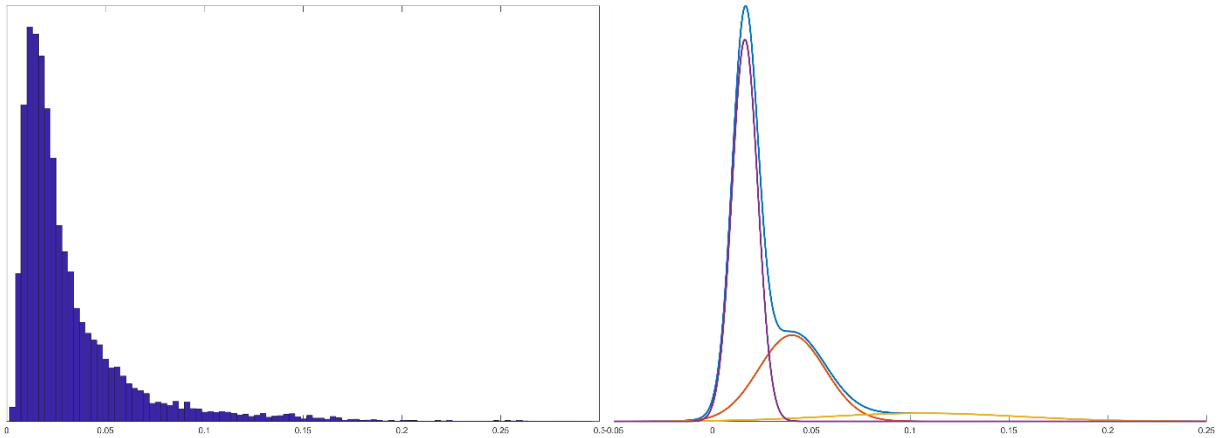

Figure 6.13: Empirical distribution of Peacock's test statistic for sample 4.

6.2.2. Merging of nearby features

After determining how to test the spatial distribution between two components and how to interpret the results, it can be decided whether to merge nearby or overlapping components or leave them separate.

For each sample, the *p*-value was calculated for a one-tailed test. The null hypothesis is that the compared features have a very similar spatial distribution. In this way, we obtain four *p*-values. To combine them into a single *p*-value and make the final decision, Fisher's method is used [66].

The Fisher method is a way of combining the results of independent tests with the same overall hypothesis. Fisher's method uses equation 12 to combine *p*-values. The value calculated using Equation 12 is then compared to the chi-squared distribution with *2k* degrees of freedom (where *k* is the number of tests) to obtain a *p*-value for combined tests.

$$\chi^2_{2k} \sim -2\sum_{i=1}^{k} \log(p_i) \tag{12}$$

The components that satisfy the null hypothesis are merged into a single component. As described earlier, the dominant component, i.e., the component described by a Gaussian distribution with the larger area, remains in the spectrum model, and its value is now a sum of both components. The other component is removed from the model. The location of the dominant component remains unchanged. We consider two components to be close if they are too close to be part of the isotopic envelope. What this means exactly is explained in the next subsection

about isotope envelope detection. This feature engineering step reduced the number of components in the spectrum model from 2884 to 2392.

6.2.3. Isotope envelope detection

A single type of a molecule can often be observed in the mass spectrum as a series of successive peaks i.e., the isotope envelope. The isotopic envelope is an expression of a specific molecule that contains different isotopes of atoms in its chemical composition, causing differences in mass and therefore differences in mass-to-charge ratio (m/z). Isotopic envelopes hinder the analysis of mass spectrum and it is beneficial to represent them as a single feature at the place of the dominant peak. Usually the difference in atomic mass between consecutive peaks is 1 Da. Peaks in an isotopic envelope should have similar shape and their spatial distribution should be the same. Using this information an algorithm for isotope envelope search was created.

By the nature of isoform envelopes the peaks of such series are uniformly distributed, with a spacing of 1 DA between them in the mass spectrum. But again, the positions of the features in the spectrum are not exact, and therefore an attempt must be made to determine the interval of values around 1 DA that is considered a valid spacing between features in an isoform envelope. A permutation test was performed between every pair of model components within the range of 1.5 DA (see Figure 6.14). The Gaussian distribution around the 1 DA distance was used to calculate the critical value. Using a two-sided test with a confidence level of 95%, the critical value is approximately 0.22 DA. This value was used in the previous subsection to calculate the threshold (1 - 0.22) for the maximum distance between nearby features.

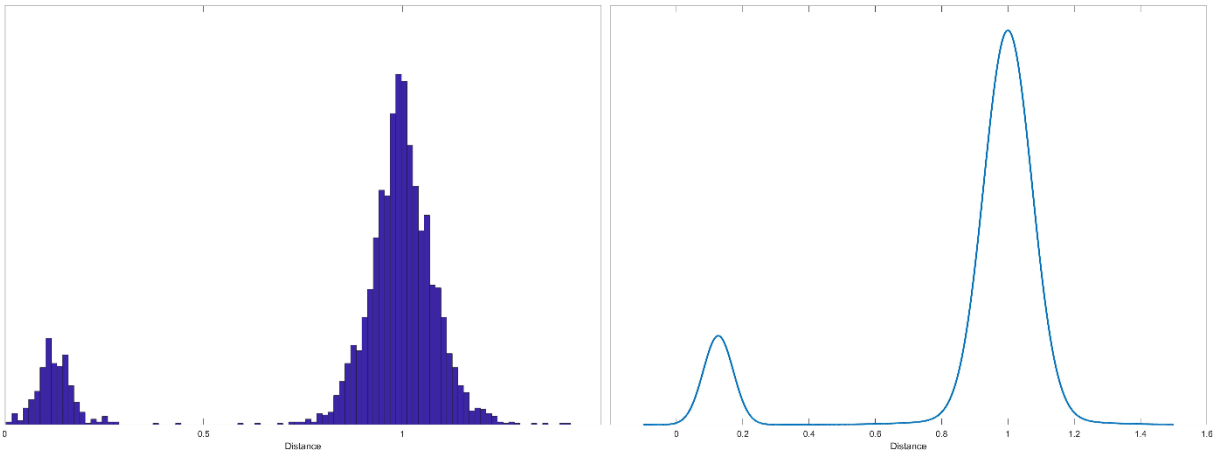

Figure 6.14: Distribution of distances between nearby features.

The second condition that two components must satisfy to be considered valid members of an isoform envelope is similarity of shape. To find the interval of valid ratio between the σ-values of the compared features, the ratios of 10 thousand random pairs of σ-values were calculated (see Figure 6.15). A two-sided test with the confidence level of 95% yielded the critical value of 0.62 for the σ ratio around one.

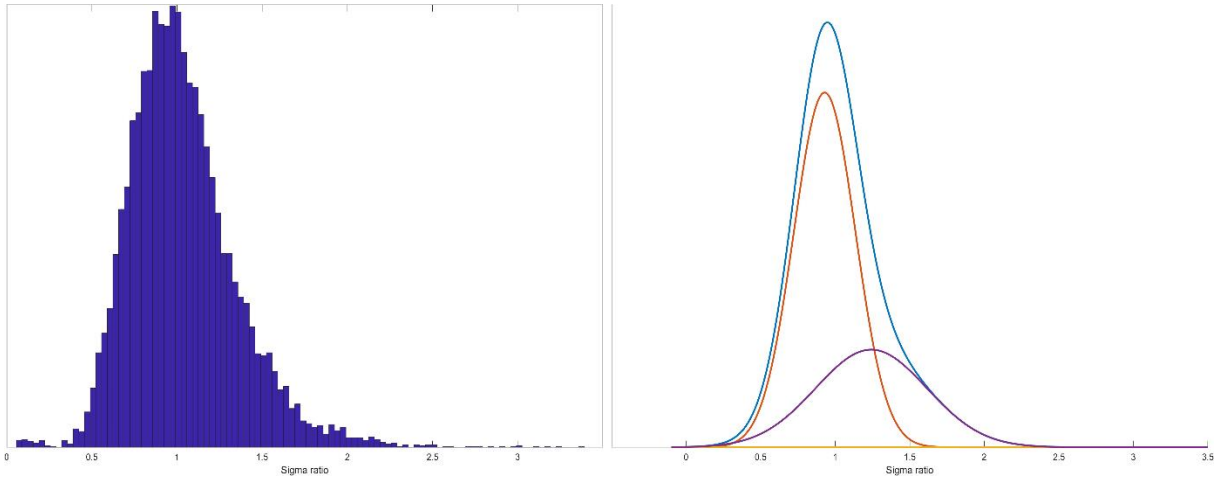

Figure 6.15: Distribution of sigma ratios.

The search for isoforms starts with the first component (lowest μ-value). Within the valid range, components are searched for. If such a component exists, the sigma ratio between the components is evaluated. If the components are within the valid range and have a similar shape, their spatial distribution is checked in the way described above. If there is no statistical difference between their spatial distribution, the components are considered to be the beginning of an isoform envelope and the search continues.

When the isotopic envelope ends, i.e., for the last element of the envelope there are no other components within the valid distance, shape, and with the same spatial distribution, the envelope is merged into a single feature. The value of this feature is the sum of the areas of all envelope components, and the location is equal to the location of the dominant component. Figure 6.16 shows an example of an isoform envelope reduced to a single feature. The dominant orange component was the beginning of the envelope, and the feature is present at the component's location. The other, blue feature is the component that was not part of the envelope.

Then, the search for the next component after the first component of the envelope continues and so on until the last component of the mass spectrum model. As a result

of this process, the remaining number of 2392 components was reduced to 888 features. The diagram of the whole process of dimensionality reduction of the data from the aggregated mass spectrum to the final feature set is shown in Figure 6.167.

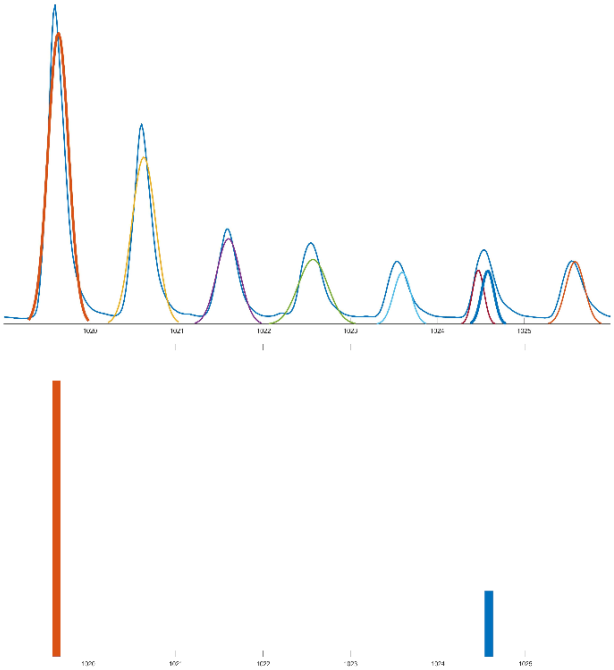

Figure 6.16: Elements of the spectrum model (top) and final features after isotope envelope detection (bottom).

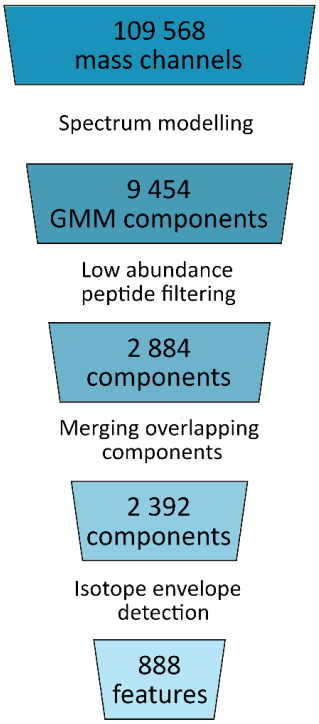

Figure 6.167: Dimensionality reduction of Mass Spectrometry data.

7. Classification

7.1. Construction of a robust classification system

The topic of applying machine learning to the data acquired by MS is well documented with many publications on the subject. Both supervise and unsupervised machine learning methods have been successfully applied for data sets acquired with MALDI-TOF MS [67, 68, 69, 70] with neural networks and logistic regression among the most successful. In most cases, classifiers achieve very good accuracy. However, the performance of classifiers highly depends on the data. The size of the data set, the preprocessing of the mass spectra, and the complexity of the sample vary greatly from case to case. Therefore, it is difficult to compare the results of different experiments, but it is clear that MS data can be used to train well performing classifiers.

We use neural networks and logistic regression to evaluate weather our spectral preprocessing workflow, particularly spectrum modeling, provides a feature set that can produce well performing classifiers. We use neural networks because of their high predictive power and logistic regression due to its interpretability. The first step is to split the data set into training, test and validation sets.

7.1.1. Splitting the data set

Before training the classifiers, the data set is divided into training, test and validation sets. 10% of the data is used as the validation set. The rest is used for the training and test sets. A single split into training and test set is used to train a single regression-based model and a neural network. The number of observations related to normal tissue is much higher than the number of observations related to epithelium or cancer. The epithelium is the least represented class with below 10% of observations labeled as epithelium. For highly imbalanced data, the training and test sets must be

appropriately chosen to ensure sufficient representation in each set. We used a standard stratified sampling strategy to select the training and test sets. For each draw, the training set consists of 70% of the remaining data and the test set consists of 30% of the remaining data.

7.1.2. Confusion matrix and performance measures

Each classifier is evaluated by applying the model to the test set. The model assigns some observations to the correct class and some to the incorrect class. The performance of the classifier is summarized by the confusion matrix. Table 7.1 shows the confusion matrix for a binary classification. Of course, the confusion matrix can be constructed for any number of classes. In this work, we model data labeled with the three classes: "cancer", "epithelium", and "normal tissue". Since the main goal of the analysis is to detect cancerous tissue, we evaluate our models by deconstructing the three-class problem into three cases of binary classification: "cancer-vs-rest", "epithelium-vs-rest", and "normal tissue-vs-rest ".

Table 7.1: Confusion matrix for binary classification.

		PREDICTED CLASS	
	All observations = Cancer + Healthy Tissue	Positive (PP) Predicted Cancer	Negative (PN) Predicted Healthy Tissue
ACTUAL CLASS	Positive (P = TP + FN) Actual Cancer	True positive (TP) Correctly classified Cancer	False negative (FN) Cancer incorrectly classified as Healthy Tissue
	Negative (N = FP + TN) Actual Healthy Tissue	False positive (FP) Healthy Tissue incorrectly classified as Cancer	True negative (TN) Correctly classified Healthy Tissue

The confusion matrix summarizes the results of the classification. To evaluate the performance of a model, the values presented in the confusion matrix are used to calculate performance measures. There are many measures that can be calculated for binary classification, but the decision about which are useful should be made on a case-by-case basis. Therefore, it is important to understand what exactly the measure

describes and how its value is affected by the characteristics of the data before making a judgment about the usefulness of the model. In what follows, we describe only the measures that were most useful in our experiments. The binary classification "cancer-vs-rest" is used as the context to explain the measures. In this case, cancer is the "positive" class and others (i.e., epithelium or normal tissue) is the "negative" class.

The first measure we calculate is the accuracy of the model, which is given by Equation 3. Accuracy is the percentage of correctly classified observations. High values for accuracy indicate that the classifier is good at predicting the correct class for a given problem. Accuracy is not an ideal measure and can be misleading for unbalanced data sets. For example, accuracy may be high due to a high rate of correctly classified observations of the majority class, even if the percentage of correctly classified observations from minority class is very low.

$$Accuracy = \frac{TP + TN}{P + N} \tag{13}$$

The second measure is precision. Precision, or positive predictive value (PPV), is the ratio of true positives to all observations that the model has classified as positive (Equation 4). In the context of a diagnostic test, high precision means that we can be confident in the positive outcome of the test. In other words, the probability of a positive result being a false positive is low.

$$Precision = PPV = \frac{TP}{TP + FP} \tag{14}$$

The third measure is the negative predictive value (NPV). The NPV describes exactly the same property as the PPV, but for the negative class instead of the positive class. It is calculated with equation 5.

$$NPV = \frac{TN}{TN + FN} \tag{15}$$

The next measure that enters into our evaluation is sensitivity, also called recall or true positive rate (TPR). Sensitivity is calculated with equation 6. For our research, a high sensitivity of the model means that there are few cases for which the model cannot detect the cancer. As sensitivity increases, the number of false positives usually increases as well.

$$Sensitivity = TPR = \frac{TP}{TP + FN} \tag{16}$$

Finally, specificity (equation 7) is the opposite of sensitivity. High specificity means that the classifier is good at detecting the negative class.

$$Specificity = \frac{TN}{TN + FP} \tag{17}$$

The above values are the most basic measures that describe the overall performance of a classifier. Although there are many more, these values are sufficient for an initial assessment in the context of a diagnostic test. It is important to remember that each classification problem should be considered separately, based on the objectives of the classification and, most importantly, on the cost of wrong decisions. There are two types of errors that a classifier can make. A type I error is a false positive, which is when the classifier assigns a positive class to an observation that is actually negative (or any other class in a multiclass classification). Similarly, an error of type II (false negative) is when the classifier fails to recognize an observation of a particular class. In our work, the task is to classify cancerous tissue, therefore that the type II error has a much higher cost than the type I error. In such a case, high sensitivity is crucial. On the other hand, if the positive result of the diagnostic test means aggressive and dangerous treatment, then the number of false positives is also important and can not be ignored. Optimization of one value has an impact on all other values. For example, perfect sensitivity of a model can be achieved by simply classifying all observations as positive. This would, of course, defeat the purpose of data modeling, but this example illustrates the danger of maximizing a single measure of classification performance.

In summary, none of the measures described can be used by themselves to evaluate the usefulness of our model. Therefore, none of them can be used to compare models with each other. For this purpose, other values and techniques are used that ensure an optimal trade-off between the different aspects of classifier performance.

One of such values is F1 score (equation 8). F1 score is the harmonic mean of sensitivity and precision. The higher the F1 score, the better the trade-off between precision and sensitivity.

$$F1 = 2 * \frac{Sensitivity * Precison}{Sensitivity + Precision} = \frac{2TP}{2TP + FP + FN} \tag{18}$$

7.1.3. Receiver operating characteristic

Although F1 score can be used as an overall measurement of a model performance a better method is to plot the receiver operating characteristic. The receiver operating characteristic (ROC) is a plot that visualizes the quality of a probabilistic model for each threshold and shows the trade-off between sensitivity and specificity of the model. The plot has 1-specificity on the x-axis and sensitivity on the y-axis (see Figure 7.1). The plot can be easily used to compare the performance of classifiers by calculating the area under the ROC curve (AUCROC or simply AUC) [71].

A number of curves have been plotted in Figure 7.1 to show how the performance of the classifier affects the representation of ROC. An ideal classifier that perfectly classifies every observation with 100% confidence is indicated in green. Such a classifier has an AUC value of 1. The red line is the no-discrimination line. It marks the worst possible classifier with the AUC equal to 0.5. The black line is an example of a more realistic curve for a model without classification capabilities. It is drawn for a classifier that randomly predicts the outcome class based on a coin toss. Finally, the blue line in Figure 7.1 represents a classifier that has some ability to discriminate between classes, although it makes quite a few errors and its overall performance is not very good.

The value of AUC may be less than 0.5, even if 0.5 is the worst value. If the AUC value is below 0.5, the classifier is able to distinguish classes, and the lower the value, the better. In such cases, the criterion for "positivity" must be reversed, and the AUC will then take values in the range of 0.5 to 1.

As we can see, for each case there are initially no type I errors. The model perfectly detects the negative class (specificity is 1), but has a low ability to detect the positive class (sensitivity is 0). As the probability threshold for placing an observation in the positive class increases, the number of true positives increases along with the number of false positives. Eventually, the roles reverse and specificity is 0 and sensitivity is 1. The shape of the curve describes how the number of true and false positives changes as the threshold changes.

At some point there is an optimal balance between sensitivity and specificity. What is an optimal balance can be defined in several ways. We chose to find the optimal threshold by maximizing the Youden index (J) (see equation 10). Geometrically, the Youden's index on the ROC curve is the vertical distance between the ROC curve and the no-discrimination line (identity line). A similar option is to maximize the absolute

distance between the ROC curve and the zero discrimination line (labeled T in Figure 7.1). In this case, the result is the same for both methods.

$$J_{max} = max_t\{sensitivity(t) + specificity(t) -\} \tag{19}$$

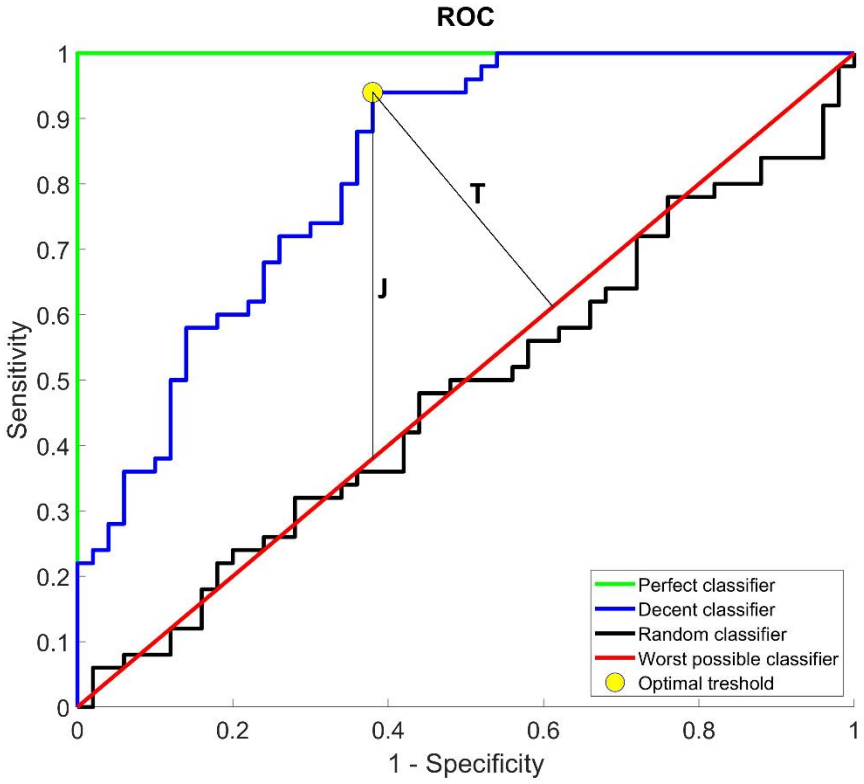

Figure 7.1: Example of ROC curves for classifiers with different predictive capabilities. *J* is the visualization of the maximum Youden's index. *T* is the maximum distance between ROC and the no-discrimination line (red line).

7.1.4. Precision-sensitivity trade-off

Although ROC is generally considered a good tool for model comparison, there is a problem when dealing with highly imbalanced data sets. The problem is that specificity is highly dependent on the number of true negatives. In very unbalanced data with a small number of positive observations, this can mask meaningful differences between classifiers. The AUC value may be deceptively high due to the high detection rate of true negatives. In such cases, the AUC value is insensitive to differences in the precision of the compared models. This is unacceptable for diagnostic tests where precision is very important. It is nevertheless useful to plot ROC and calculate AUC values for an overall assessment of models.

For a better visual and numerical comparison of models trained on imbalanced data, the trade-off between precision and sensitivity can be plotted. The precision-sensitivity plot is very similar to the ROC plot. On the y-axis it has precision and on the x-axis it has 1-sensitivity (see Figure 7.2). Precision-sensitivity plots are more resilient to data sets with a small number of positive classes, since neither precision nor sensitivity is affected by high numbers of true negatives. Since the high rate of true negatives does not affect the plot, the area under the precision sensitivity curve is a better measure for comparing such data sets.

There is a crucial drawback to the precision sensitivity curves. While the baseline of the ROC curve is always the same, the baseline of the precision sensitivity curve depends on the balance between the positive and negative classes. Figure 7.2 A shows the precision-sensitivity curve for a perfectly balanced data set where the positive and negative classes are equally represented in the data set. In such a case, as with ROC, the worst possible AUC value is 0.5 and the best is 1. The situation changes when there are more observations of one class. Figure 7.2 B shows the precision-sensitivity plot for the data set where only 25% of the data is the positive class. In such a case, the minimum value for the AUC is 0.25.

In summary, the ROC curves can be used to compare any pair of classifiers, whereas the precision-sensitivity curve can only be used to compare models trained on data with similar proportions of positive and negative classes.

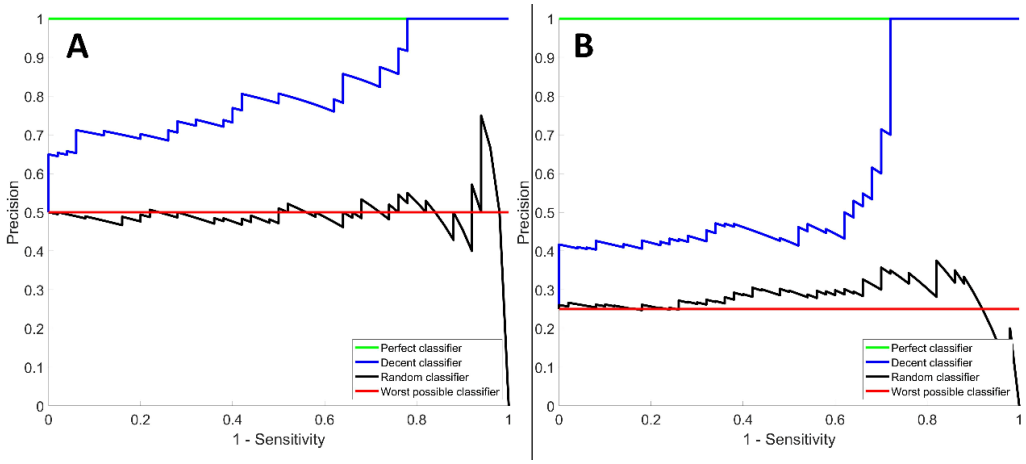

Figure 7.2: Precision-sensitivity plot for a model trained on data set with balanced classes (A) and a model trained on data with ¼ ratio of positive and negative classes (B).

7.2. Multinominal regression-based classifier

Training the classifier using multinomial regression is done iteratively by adding features to the list of predictors used for multinomial regression until the stop condition is reached. One of the goals, of course, is to train the best possible classifier, but at the same time the most valuable features are identified, and thus, potentially biomarkers, that are directly correlated with the presence or absence of the disease. The algorithms are run multiple times to mitigate the randomness of the division into training and test sets.

Features are selected sequentially and added to the list of predictors used for regression in future iterations. Selection of the best feature in the current iteration of the algorithm is done by running regression on the training set separately for each feature (along with the features already in the list of predictors). The feature that, when added to the predictor list, produces the model with the best likelihood is selected as the best feature. Then the process is repeated until the stop condition is satisfied. The algorithm stops when the Bayes factor indicates a very strong similarity between the models before and after adding a new feature to the predictor list. The Bayes factor is calculated based on the Bayesian information criterion computed for models with and without the new feature. As explained in previous chapters, BIC introduces a penalty for model complexity. Such a stopping condition protects the trained models from overfitting.

Classification is done by assigning the class to which the model gives the highest probability for the given observation. In classification, the main goal of this work is to assess whether a good diagnostic test can be performed. For this reason, this multiclass classification is transformed into a binary classification in a one-vs-rest strategy. We are particularly interested in the performance of the cancer-vs-rest binary classification. The full results are shown in Table 7.2.

Figure 7.3 shows how the ROC curve changes as new features are added to the list of predictors for multinomial regression, and Figure 7.4 illustrates how the precision-sensitivity curve changes as new features are added to the list. As we can see, the regression with a single predictor has very poor results, but the model trained with three features already has a decent ability to predict the cancer class.

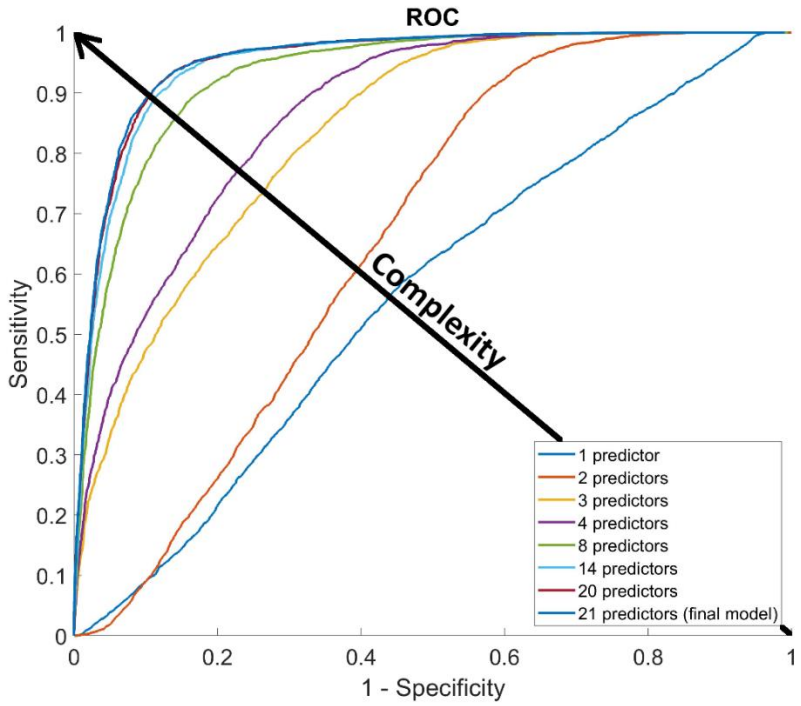

Figure 7.3: Influence of model complexity (number of regression predictors) on the ROC curve of the classifier.

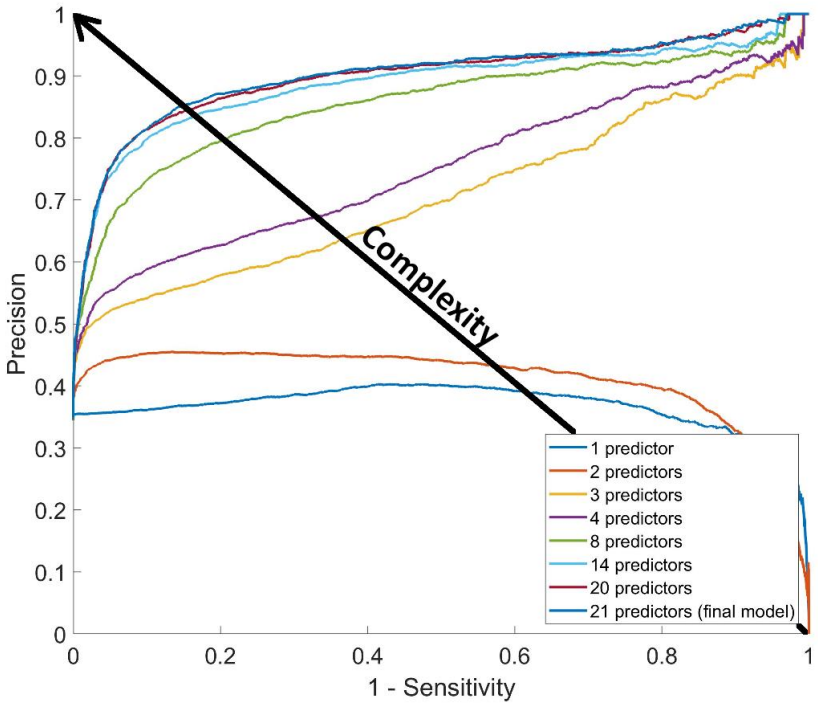

Figure 7.4: Influence of model complexity (number of regression predictors) on the precision-sensitivity curve of the classifier.

After several runs of the algorithm, each time with a new random division into training and test sets, we have the following results. The final complexity of the models varies. The simplest model has 16 predictors and the highest number is 27.

Figure 7.5 shows ROC and precision sensitivity curves for ten randomly selected models trained with the algorithm. Initial assessment can be drawn that multinomial regression based classifiers work well with the data processed with our workflow. Comparing the two plots, we can also notice the differences between the ROC and the precision-sensitivity curves. The differences between models are more visible in the precision-recall curve.

Coefficients for multinomial regression were calculated using the MATLAB function "mnrfit". The training process took several hours using an eight-core Intel(R) Core(TM) i7-11700K processor. The training time is considerable, but it is proof of concept that the proposed dimensionality reduction and feature engineering process, which reduces the hundreds of thousands of data points to less than a thousand features is able to extracts the most important patterns from the data.

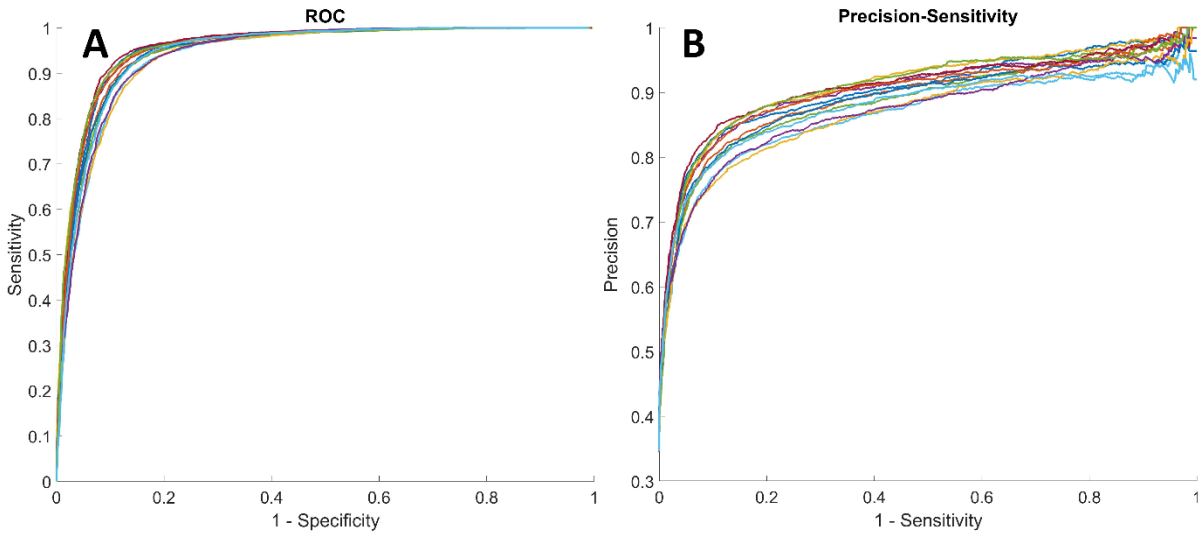

Figure 7.5: ROC (A) and precision-sensitivity (B) curves for randomly selected 10 models trained with multinominal regression-based algorithm.

Assigning the class with the highest probability score given by the regression model leads to models with good performance with mean accuracy above 90% and mean precision above 90%. Low specificity for epithelium-vs-rest classification is the result of unbalanced data. The measurements for all binary classifications in one-vs-rest strategy are presented in the Table 7.2. The values are the mean values from all trained models with the 95% confidence intervals for the mean. The last column presents the averaged results from all three binary classifications. AUC_{ROC} is the area under the ROC curve and AUC_{PS} is the area under the precision-sensitivity curve.

Table 7.2: Mean performance measures for multinominal regression models with 95% confidence intervals.

Measure [%]	Caner-vs-rest	Epithelium-vs-rest	Normal tissue-vs-rest	Overall mean
Accuracy	**88.70** [87.99 ; 89.42]	**96.88** [96.81 ; 96.94]	**86.98** [86.29 ; 87.67]	**90.85** [90,37 ; 91,34]
Precision (PPV)	**91.34** [90.75 ; 91.94]	**97.29** [97.20 ; 97.38]	**85.72** [84.80 ; 86.63]	**91.45** [90.97 ; 91.93]
NPV	**83.68** [82.55 ; 84.82]	**85.71** [84.65 ; 86.76]	**87.76** [87.12 ; 88.41]	**85.72** [85.19 ; 86.24]
Sensitivity	**91.44** [90.79 ; 92.09]	**99.46** [99.41 ; 99.51]	**81.03** [79.95 ; 82.12]	**90.65** [90.15 ; 91.14]
Specificity	**83.49** [82.29 ; 84.69]	**53.72** [52.20 ; 55.24]	**90.96** [90.35 ; 91.58]	**76.06** [75.13 ; 76.98]
F1 score	**91.39** [90.84 ; 91.94]	**98.36** [98.33 ; 98.40]	**83.30** [82.39 ; 84.21]	**91.02** [90.53 ; 91.51]
AUC_{ROC}	**94.90** [94.46 ; 95.35]	**96.71** [96.45 ; 96.96]	**93.98** [93,52 ; 94,43]	**95,20** [94.85 ; 95,54]
AUC_{PS}	**89.10** [88.11 ; 90.10]	**78.56** [77.70 ; 79.42]	**96.20** [95.92 ; 96.49]	**87,96** [87,31 ; 88,61]

Classifiers can be further improved by balancing the trade-off between performance measures. As mentioned, for ROC it can be done by maximizing the Youden's index. Maximizing the Youden's index for ROC curve optimizes the

balance between sensitivity and specificity. With the same strategy other pairs of measures can be balanced. Threshold can be chosen for example for the balance between positive and negative predictive values. Each binary classifier was optimized for PPV-NPV trade-off by maximizing the distance between the curve and no-discrimination line, analogically to Youden's index (see Figure 7.6).

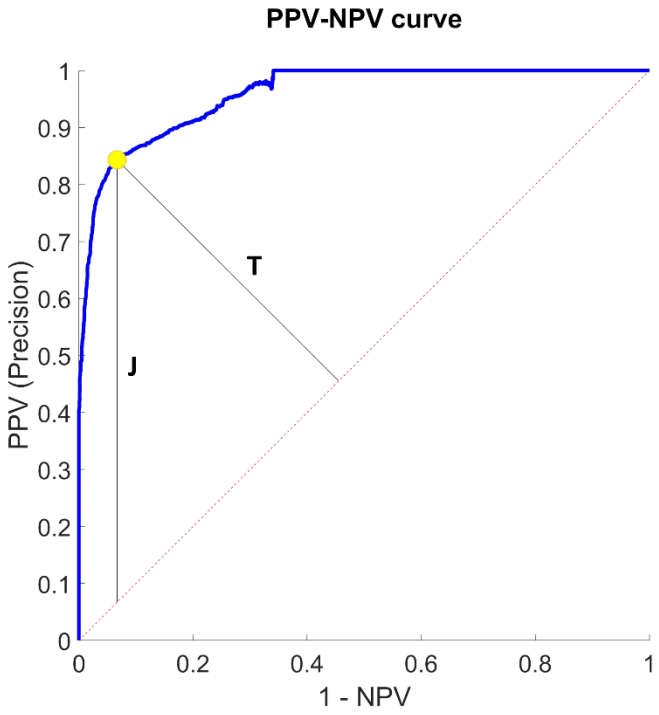

Figure 7.6: Threshold optimization using PPV-NPV curve. *J* is analogical to the Youden's index for ROC curves. *T* is the distance from no-discrimination line.

After calculating the threshold the binary classification is done by assigning the positive class to the new observation if the probability of that class in the multinominal regression model is higher then the threshold. Results are presented in the Table 7.3.

Table 7.3: Mean performance measures for multinominal regression models after balancing PPV and NPV with 95% confidence intervals.

Measure [%]	Caner-vs-rest	Epithelium-vs-rest	Normal tissue-vs-rest	Overall mean
Accuracy	**89.13** [88.46 ; 89.79]	**94.84** [94.71 ; 94.97]	**87.73** [87.13 ; 88.34]	**90.57** [90.13 ; 91.00]
Precision (PPV)	**92.47** [91.99 ; 92.96]	**94.82** [94.70 ; 94.94]	**83.66** [82.32 ; 84.99]	**90.32** [89.78 ; 90.86]
NPV	**83.15** [81.73 ; 84.58]	**97.55** [96.72 ; 98.39]	**90.69** [90.04 ; .91.34]	**90.47** [89.73 ; 91.20]
Sensitivity	**90.82** [89.89 ; 91.74]	**99.99** [99.99 ; 99.99]	**86.37** [85.22 ; 87.52]	**92.39** [91.92 ; 92.86]
Specificity	**85.91** [84.92 ; 86.91]	**08.78** [06.48; 11.09]	**88.65** [87.45 ; 89.84]	**61.12** [60.08 ; 62.15]
F1 score	**91.63** [91.09 ; 92.17]	**97.34** [97.27 ; 97.40]	**84.96** [84.28 ; 85.63]	**91.31** [90.91 ; 91.71]
AUC_{ROC}	**94.90** [94.46 ; 95.35]	**96.71** [96.45 ; 96.96]	**93.98** [93,52 ; 94,43]	**95.20** [94.85 ; 95.54]
AUC_{PS}	**89.10** [88.11 ; 90.10]	**78.56** [77.70 ; 79.42]	**96.20** [95.92 ; 96.49]	**87.96** [87.31 ; 88.61]

7.3. Neural network classifier

Neural networks are a great tool for pattern recognition and are used in numerous fields, providing high efficiency and flexibility. Neural networks are capable of

handling the most difficult tasks depending on their type, architecture and complexity. While the regression-based algorithm required several hours to train models, a simple neural network trained a model within a few minutes. Figure 7.7 shows the ROC and precision-sensitivity curves for the models trained with the same training sets and evaluated on the same test sets that were used for multinomial regression. The neural network architecture is very simple and consists of two hidden layers, each with the same number of nodes as the number of features in the data set. The MATLAB implementation "patternnet" of the neural network was used for model training. The performance of the classifiers is very good, although not as good as achieved by the multinomial regression-based algorithm. The results for all binary one-vs-rest classifiers presents Table 7.4. Table 7.5 shows the performance of the models after balancing PPV and NPV.

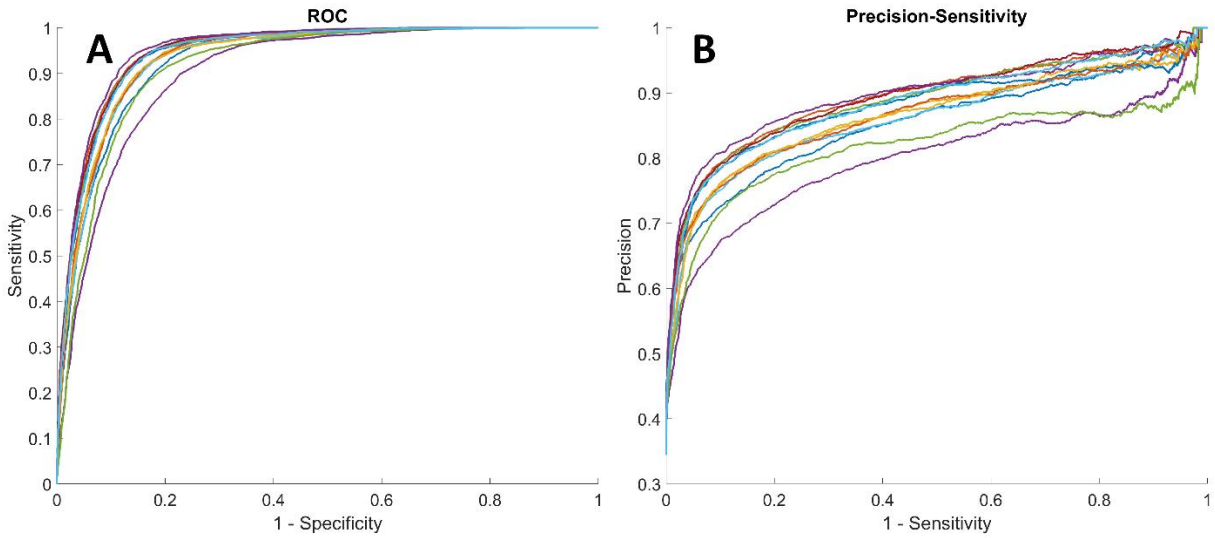

Figure 7.7: Description. ROC (A) and precision-sensitivity (B) curves for randomly selected 10 models trained with neural networks.

Table 7.4: Mean performance measures for neural network models with 95% confidence intervals.

Measure [%]	Caner-vs-rest	Epithelium-vs-rest	Normal tissue-vs-rest	Overall mean
Accuracy	**86.06** [84.89 ; 87.24]	**96.74** [96.60 ; 96.89]	**84.92** [83.77 ; 86.07]	**89,24** [88,43 ; 90,05]
Precision (PPV)	**88.07** [86.70 ; 89.44]	**97.05** [96.90 ; 97.20]	**85.38** [84.70 ; 86.06]	**90,17** [89,48 ; 90,86]
NPV	**81.86** [80.81 ; 82.92]	**87.65** [85.91 ; 89.38]	**84.71** [83.28 ; 86.14]	**84.74** [83.69 ; 85.79]
Sensitivity	**91.14** [90.76 ; 91.52]	**99.58** [99.50 ; 99.65]	**75,22** [72,45 ; 77,99]	**88,65** [87,68 ; 89,62]
Specificity	**76.39** [73.31 ; 79.47]	**49.39** [46.72 ; 52.07]	**91.41** [91.12 ; 91.70]	**72.40** [70.61 ; 74.18]
F1 score	**89.57** [88.77 ; 90.37]	**98.30** [98.22 ; 98.37]	**79.93** [78.09 ; 81.77]	**89.27** [88.37 ; 90.16]
AUC_{ROC}	**93.76** [92.99 ; 94.53]	**95.92** [95.03 ; 96.80]	**93.52** [92.95 ; 94.09]	**94.40** [93.69 ; 95.11]
AUC_{PS}	**86.72** [85.16 ; 8829]	**74.54** [71.65 ; 77,.42]	**96.06** [95.69 ; 96.42]	**85.77** [84.23 ; 87.31]

Table 7.5: Mean performance measures for neural networks models after balancing PPV and NPV with 95% confidence intervals.

Measure [%]	Caner-vs-rest	Epithelium-vs-rest	Normal tissue-vs-rest	Overall mean
Accuracy	**85.73** [82.11 ; 89.35]	**94.62** [94.49 ; 94.75]	**87.09** [86.26 ; 87.92]	**89.15** [87.71 ; 90.59]
Precision (PPV)	**89.43** [85.11 ; 93.75]	**94.61** [94.49 ; 94.74]	**80.71** [79.13 ; 82.29]	**88.25** [86.46 ; 90.05]
NPV	**81.69** [78.75 ; 84.86]	**94.67** [91.64 ; 97.70]	**92.29** [91.51 ; 93.07]	**89.55** [88.17 ; 90.93]
Sensitivity	**90.00** [87.96 ; 92.05]	**99.99** [99.99 ; 99.99]	**89.27** [88.03 ; 90.51]	**93.09** [92.41 ; 93.76]
Specificity	**77.59** [63.72 ;]	**04.95** [02.62 ; 07.29]	**85.63** [84.07 ; 87.19]	**56.06** [51.00 ; 61.12]
F1 score	**89.42** [87.54 ; 91.29]	**97.23** [97.16 ; 97.29]	**84.73** [83.87 ; 85.59]	**90.46** [89.57 ; 91.35]
AUC_{ROC}	**93.76** [92.99 ; 94.53]	**95.92** [95.03 ; 96.80]	**93.52** [92.95 ; 94.09]	**94.40** [93.69 ; 95.11]
AUC_{PS}	**86,72** [85,16 ; 88,29]	**74,54** [71,65 ; 77,42]	**96,06** [95,69 ; 96,42]	**85,77** [84,23 ; 87,31]

7.4. Multinominal regression vs neural network

Finally, we compared both algorithms by classifying the validation set with models without PPV-NPV balancing. Figure 7.8 shows the ROC and precision sensitivity

curves for the averaged results of each method with 95% confidence intervals (for the binary classification cancer-vs-rest). The multinomial regression-based method is shown in blue and the neural networks are shown in yellow. As we can see, the multinomial regression-based classifier has better overall performance than the neural networks. The performance measures for the validation set classification are shown in Table 7.6.

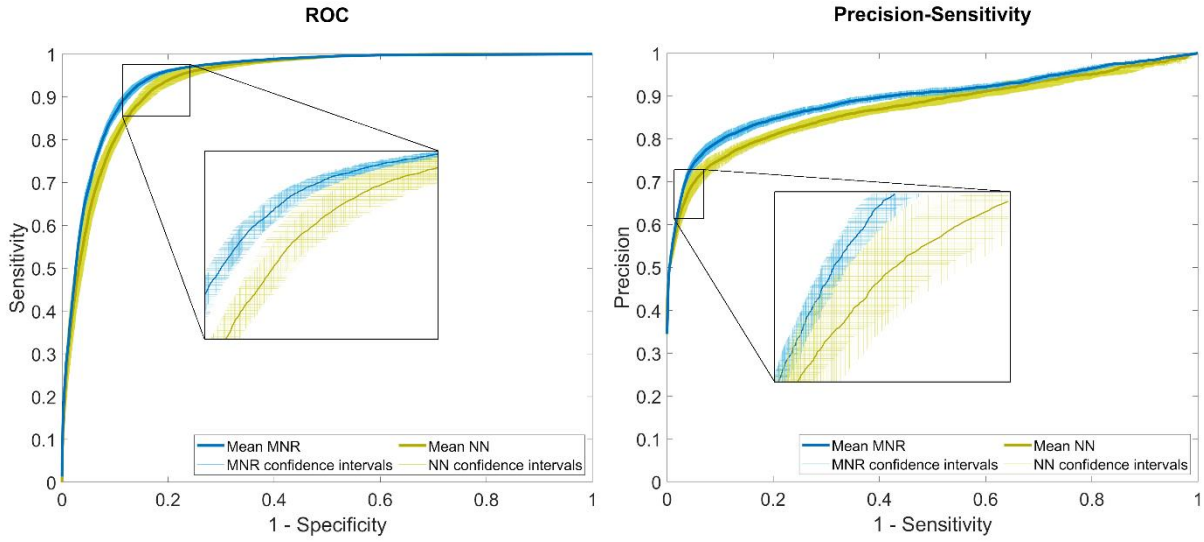

Figure 7.8: ROC (left) and precision-sensitivity (right) curves for neural networks (yellow) and multinominal regression (blue) with 95% confidence intervals.

Table 7.6: Comparison of performance for cancer-vs-rest classifiers on validation set.

Measure [%]	Multinominal regression	Neural networks
Accuracy	**88.51** [87.78 ; 89.23]	**86.07** [84.94 ; 87.19]
Precision (PPV)	**91.54** [90.88 ; 92.20]	**88.18** [86.90 ; 89.46]
NPV	**82.87** [81.87 ; 83.87]	**81.68** [80.58 ; 82.79]
Sensitivity	**90.88** [90.33 ; 91.42]	**90.99** [90.56 ; 91.43]
Specificity	**83.99** [82.69 ; 85.30]	**76.69** [73.84 ; 79.53]
F1 score	**91.20** [90.66 ; 91.75]	**89.55** [88.78 ; 90.33]
AUC_{ROC}	**94.91** [94.45 ; 95.38]	**93.77** [92.89 ; 94.65]
AUC_{PS}	**89.42** [88.44 ; 90.40]	**87.18** [85.47 ; 88.90]

7.5. Feature scoring

A machine learning model is interpretable if some of its properties can be understood by a human [72]. In other words, a model is interpretable if we can judge how the input affects the output of the model. There are interpretable methods that can always be understood by humans at some level, for example decision trees or linear models. Decision trees can be represented as a set of "if, then, else" rules that a human can understand. Linear models assign weights to each feature so the impact of a feature can be easily compared with another feature by a human. With so-called "black-box" models, the interpretability of a machine learning model can be difficult to achieve, because there is no easy way to figure out how much of an impact a feature has on the outcome. The topic of interpretable (or explainable) machine learning is well documented with descriptions of model-independent methods that can be used for any machine learning model [73] and model-specific methods developed to interpret the results of a particular algorithm. Model-specific methods for interpretable neural networks [74] attempt to explain image classification using neural networks and are not useful for our application.

The importance of interpretability cannot be overstated when it comes to diagnostic tests. Being able to make a diagnosis based on a sample is crucial, but ultimately the reasons for the diagnosis are most important. Only when we know what causes the disease can we begin to develop appropriate drugs. In recent years, both global and local methods for interpreting "black box" machine learning methods have been developed, and new approaches have emerged. Here, we propose feature scoring systems for our two classification algorithms based on well-established methods for interpreting ML models.

For our multinomial regression-based algorithm, assigning a score to a feature is not a problem. The final product of the algorithm is an ordered list of features. On this basis, it is easy to create some kind of scoring system. In this case, we simply assign features a score equal to 1/x, where x is the position of the feature in the list. In this way, we emphasize the influence of the first few features on the overall score. The total score of the features is the average score from all trained models.

For neural networks, the task is not so simple. Although neural networks are based on simple mathematical operations, the neurons are interconnected with nonlinear activation functions through several hidden layers. This means that there is no simple mathematical expression that can explain the influence of a feature on the result. For this type of methods, interpretation tools must be used.

The interpretability of a model is a spectrum. The outcome may be the result of numerous interacting factors. We may have all the knowledge about the decision process or only a few facts. Interpretation methods provide the knowledge in the form of feature statistics, visualizations, or entire models that explain the model under study [75]. In this work, we are satisfied with knowing the relative importance of the features and obtaining a numerical score to rank the features by importance.

7.2.1. LIME

One of the model-agnostic methods is the local interpretable model-agnostic explanation (LIME). LIME is an algorithm that can explain the predictions of any classifier or regressor by approximating it locally with an interpretable model [76]. LIME is a local method, which means that it explains the decision behind a single observation. Using the observation LIME creates a new data set, by perturbing the feature values to create new observations. The new value for a feature is drawn from its normal distribution, and the mean and standard deviation for that distribution are calculated based on the entire data set or a local neighbourhood. In this work, we used the entire data set. Then, new observations are assigned a weight based on their similarity to the observation under study. Each new observation is also given a prediction using the black box model that we are trying to explain. This labeled data set is then used to train an interpretable model such as a decision tree, linear regression, or other interpretable method. Finally, using this interpretable model, inferences can be made about the original observation.

In this work we try to find important features for the whole model and not just for a single observation. To do that we run the LIME algorithm a few thousand times for randomly selected observations. The simple interpretable model trained on perturbed observations were decision trees. Each time, the LIME algorithm returns an ordered list of the most important features. The feature are then scored in the same way as in our regression-based method. Repeating the LIME algorithm thousands of times provides insight into the global interpretation of the model. The advantage of this approach over using a global model-agnostic model is that we can focus on a specific type of observation, for example, by applying LIME only to observations labeled as cancer.

LIME algorithm is not ideal, new observations are made without considering correlations between features, the results depend on the definition of the local neighbourhood, which is difficult with tabular data and requires the choice of

a parameter value that has a significant impact on the result. The results also change slightly each time the algorithm is run. The stochastic nature of this algorithm is mitigated by averaging over many algorithm runs, but the results presented are only an estimate of feature importance for our neural network models [75].

7.2.2. Shapley Values

Another way to evaluate the importance of features in a black-box model is to use Shapley values [77]. The idea behind Shapley values is that features interact to provide an outcome, and the contribution of a feature can be calculated by examining how the prediction changes on average when we predict the outcome with and without that feature contribution. A detailed mathematical description of how Shapley values are calculated can be found here [78].

In simple terms, the contribution of a feature is calculated for a given subset of features. The overall score is calculated with and without the feature to determine the contribution of that feature to the overall score. The Shapley value of a feature is the averaged contribution of the feature over all possible feature sets. To obtain a prediction using the black box model, all features must have a value, so a feature cannot simply be removed. A subset of features also cannot be used for prediction. Therefore, the absence of a feature is simulated by drawing random instances from the data set and averaging results.

To calculate the exact Shapley value of a feature, all possible sets of features with and without that feature must be calculated. Since the number of all possible sets increases exponentially with the number of features, implementations of this algorithm compute estimates of Shapley values by limiting the maximum number of feature subsets.

Shapley values the same as LIME give the interpretation of a single observation. For global feature importance the Shapley values were calculated for multiple observations. Dude to higher computation time the number of runs was an order of magnitude smaller then for LIME algorithm. The final score of a feature is calculated based on the position on the importance list in the same way as for previous methods.

7.2.3. Best features

Figure 7.9 to Figure 7.12 show the spatial distribution of the top 5 features with the highest scores for the multinomial regression-based model and the neural network computed using both the LIME method and Shapley values.

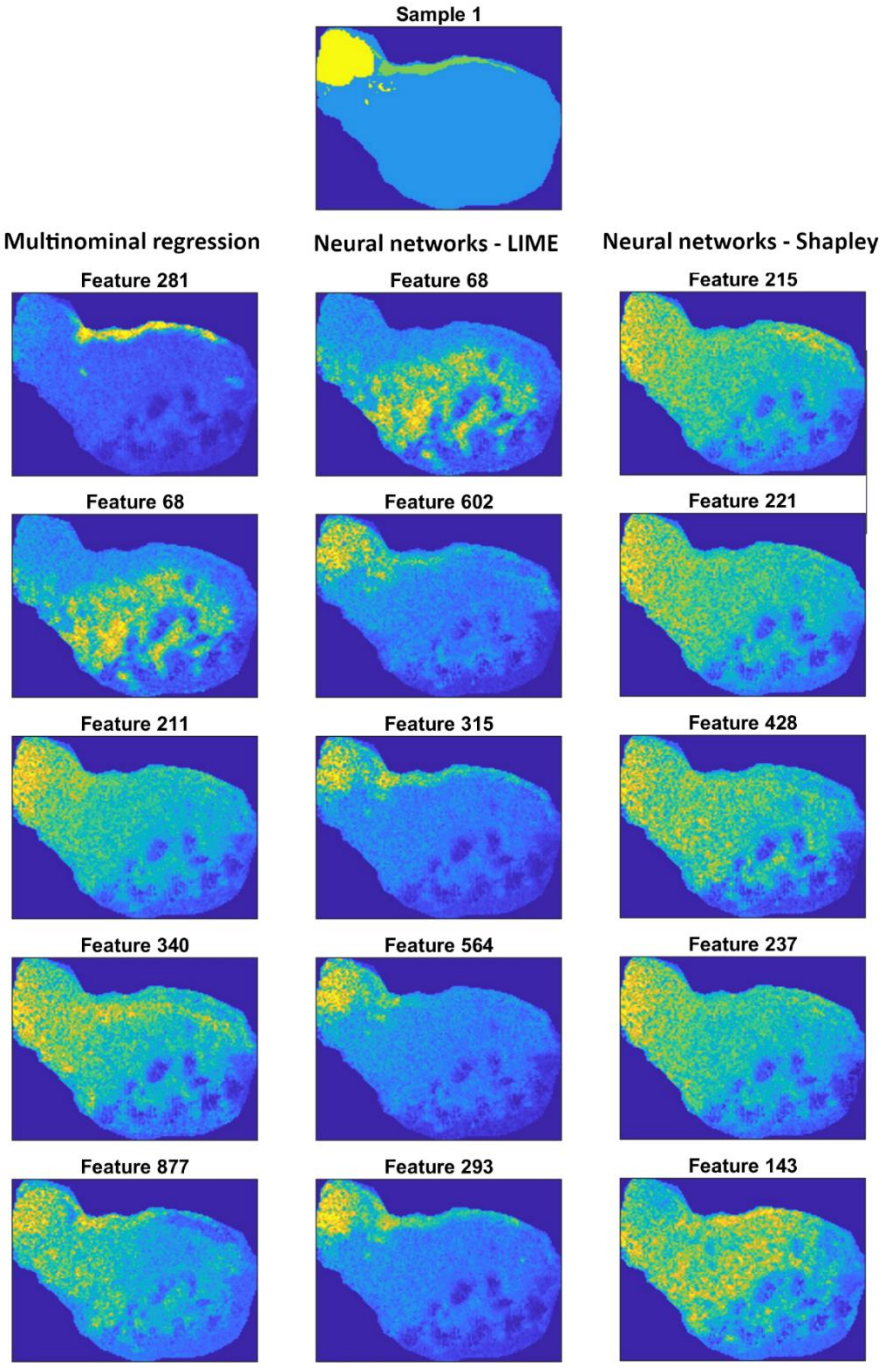

Figure 7.9: Spatial distribution of best features on the sample number 1.

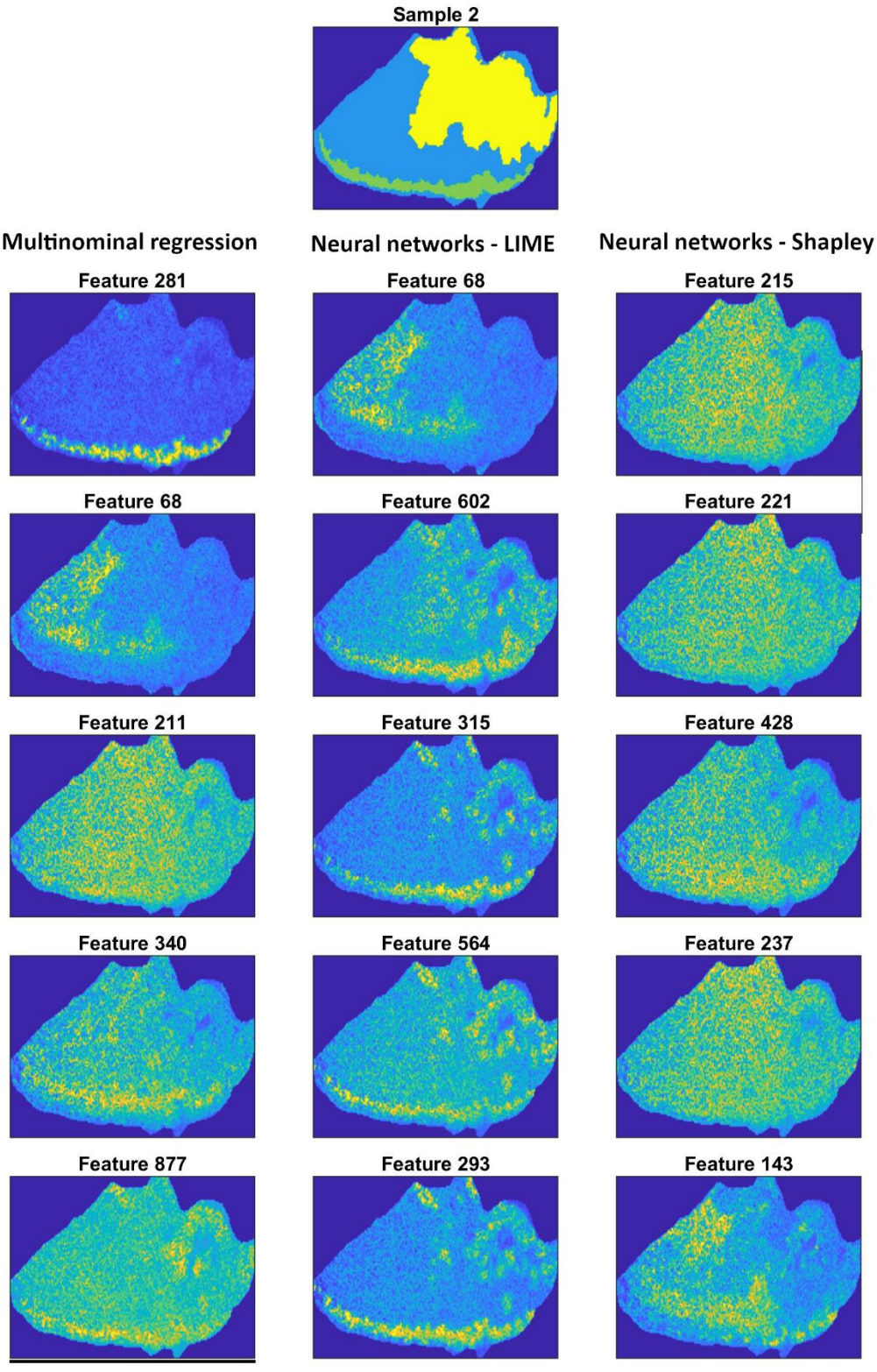

Figure 7.10: Spatial distribution of best features on the sample number 2.

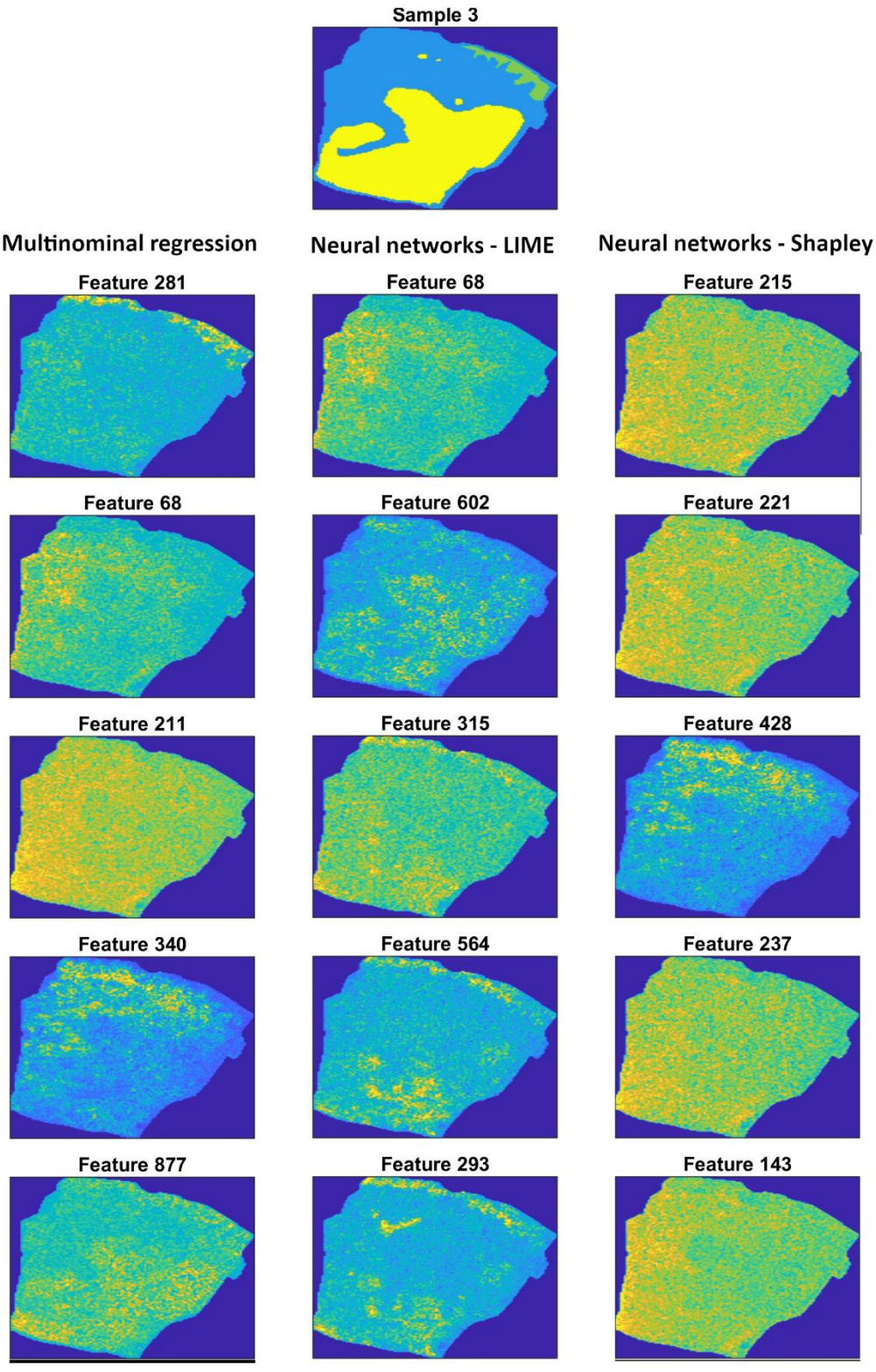

Figure 7.11: Spatial distribution of best features on the sample number 3.

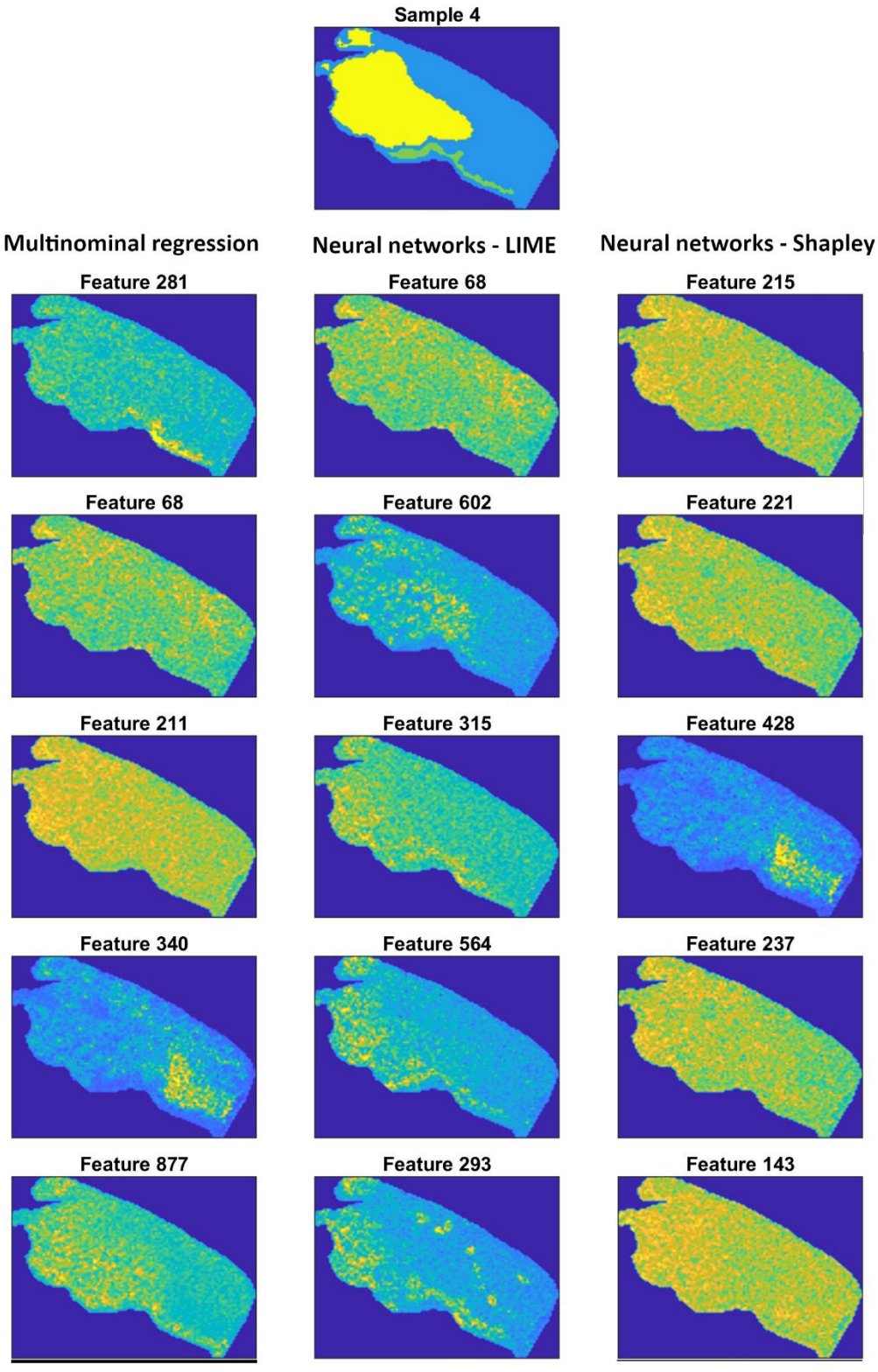

Figure 7.12: Spatial distribution of best features on the sample number 4.

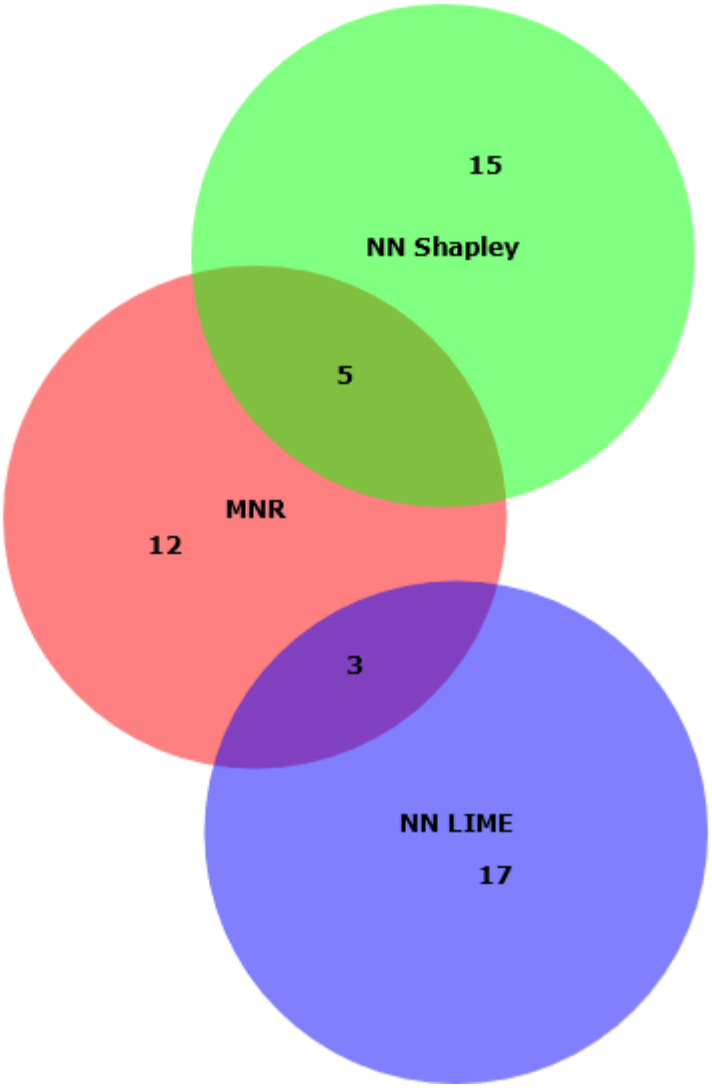

Figure 7.13: Venn diagram of top 20 features.

Figure 7.13 shows the Venn diagram of the top 20 features of the multinomial regression-based model and the neural network models computed using both LIME and Shapley values. As we can see, some of the best features are used by both methods. LIME and Shapley values computed for the same models have completely different features. The likely reason for this is that due to the computational complexity of Shapley values for large features and data sets, we only compute estimates of Shapley values for local interpretations. In addition, multiple runs of the algorithm must be performed to obtain the global feature values. Because of the long computation time, the Shapley values were computed for only a few hundred

observations and for a sample of the training set. It seems that the sample size and number of local interpretations is too small to provide meaningful results. The same conclusion can be drawn from visual inspection of Figure 7.9 through Figure 7.12. The spatial distribution of the top features obtained with the Shapley method appears to be much more uniform than with regression and LIME.

8. Conclusions

Processing of mass spectrometry imaging data is a complicated, multi-step process. As the apparatus for mass spectrometry and techniques for the data analysis evolve, the potential for the further research expands. It is certain, that the topic of MS and MSI data analysis will be further researched in years to come. During the creation of this paper, we have explored a number of different ideas and approaches to the task, and came up with a solution that in our opinion is superior to the most state-of-the art methods, and has a great potential for further improvement.

The results show that extensive and well organized analysis based on a statistical approach can provide a concise and meaningful feature set and confirm the validity of the first hypothesis stated in the dissertation. Spectrum modeling presented in the chapter 5 and our approach to feature engineering described in the chapter 6 provided very good results. There were many challenges that we had to overcome to arrive at the final form of the data processing workflow and we learned a lot during the experiments. It is our belief that there is potential for further improvement of both steps.

The number of peaks after noise filtering and redundancy removal is a few hundred. The number is at the level that was expected based on our knowledge about the number of distinct molecules in such complex biological samples as the one we examined. Not only the number of the features in the final set, or the very good performance of classifiers trained on the processed data, speaks for the quality of the workflow. Also the visual inspection of the spectrum model supports the thesis. Most of the remaining peaks are clearly correlated with a region of the spectrum where a human can reasonably assume that a true peak exists. This and other performed experiments, in our opinion, prove the superiority of peak detection methods that take into account the shape of the peak.

Our classifiers achieved very good performance with over 90% overall accuracy and over 95% AUC_{ROC}. These results, especially with such extensive data processing workflow are very convincing. It is clear that the goal of reducing the dimensionality

and redundancy was achieved and the second stated hypothesis is true, as the comparison of spatial distribution had a crucial role in the process. The removal of redundancy and dimensionality reduction by comparing the spatial distributions of components and by identifying the isotope envelopes retained the valuable information hidden in the data and didn't hinder the data potential to discriminate between classes.

Based on our investigation of peak detection method we conclude that using more complex peak detection methods like spectrum modeling, the full potential of mass spectrometry data can be explored, while simultaneously reducing the volume of the data to manageable size and retaining the valuable information about underlying biological features. Simple signal-to-noise ratio-based peak picking methods simply can not deal with mixtures of high complexity without substantial information loss.

The experiments with splitting the mass spectrum into parts before modeling with Gaussian mixtures, convinced us that it is a crucial part of spectrum modeling. Although, proposed method proved very successful for our data, further analysis is necessary to examine the impact of other baseline-correction methods on this approach. The defined rules for successful division are simple and can be used to arrive at different solution.

By trying different approaches, we learned that the most effective way to remove noise from the MSI data when using spectrum modeling, is to remove it after the model has been acquired. Filtering spectrum model elements based on their parameters rather than just peak intensity proved to be effective. Of course, denoising can be also performed before the spectrum modeling or not at all but we believe, based on our experiments, that the solution proposed in this work is the best approach.

When it comes to feature engineering, other methods can be considered, such as classical feature selection by machine learning, e.g., based on the variance of the features or by removing highly correlated features. The decision to remove the excess of features, by examining the distribution of model parameters was dictated by the desire to preserve as much interpretability as possible. If we assume that there are 10 features that are very highly correlated and very useful for classification, a classifier would normally use only one of them. To the classifier, the other features have no additional value, but from a medical research perspective, all 10 features are equally important.

It is hard to predict the impact of various factors that influence the data acquisition process on the ability of our classifiers to predict the classes of new data. Using a different mass spectrometer, samples being prepared by a different person (even assuming the same procedures) and various other factors influence the mass spectra. It

is, however, important to remember that our data set already consists of four samples and considering the excellent results, we expect our classifiers to achieve at least good performance for samples taken from new patients.

The statistical approach to classification using multinomial regression proved to be very effective. The excellent performance was at the cost of quite long computation time but the results are rewarding. Moreover, only the training of the model is time consuming, the classification of new records is instant. That means it could be used as a tool for diagnosis. Another great feature of this method is the interpretability. The feature importance can be assessed very fast. Images of best features clearly are correlated with molecules present or absent in the regions corresponding to a specific class. It is another argument in favour of both first and second hypothesis.

On the other had, we have a completely different approach to classification. The neural networks were very fast, as we intentionally used a simple architecture of the network. The results again, prove that the workflow was successful, as the model performance is also very good. However, attempts to identify the most important features in a black box model that is neural network, show how important is the interpretability of prediction models. Using LIME and Shapley values we calculated the scores for features and ordered them by importance. The two methods applied for the same models gave completely different results for the top features. None of the top 20 features were the same. However, after visual inspection of the images, it appears that the top features identified by calculating Shapley values are not that great. Shapley values had to be calculated for a very limited subset of training sets and the number of local experiments to reason about global trends was clearly too small. Maybe calculating exact Shapley values would provide a better results but for the reasons explained in the thesis it was impossible. LIME-based scoring, on the other hand, provided viable results, although from the top 20 features identified for both multinominal regression-based classifiers and neural networks, only 3 features are the same. Separately for both methods, with the exception of a few features that were always among the top performers, the remaining features changed greatly for each instance of a unit model. This is the result of the heterogeneous nature of the data. By aggregating the results of many models trained on different training sets, we were able to determine all the top features regardless of which subset in the data was most influential for a given training set. The final top features change only slightly after we repeat the entire process, of global feature importance calculation. This confirms the final hypothesis of the work.

In the future we would like to continue our work with MSI data analysis particularly by further improving the spectrum modeling and feature engineering process. We would also like to apply our method to other existing data sets and to new data. Further research will also involve comparing the method with other novel methods. Finally, we want to use the results of our data processing and feature scoring to identify biomarkers correlated with the squamous cell carcinoma of the head and neck.

Abstract

The subject of the dissertation is the analysis of data acquired by mass spectrometry imaging of samples obtained from patients with head and neck cancer. The following hypotheses were made in the thesis. The first hypothesis states that peak identification in mass spectra can be successfully performed using a spectrum modeling approach by fragmenting the spectrum into parts and then modeling them with Gaussian mixture models. The second hypothesis states that the spatial distribution information obtained through imaging can be used to remove redundancy and reduce the dimensionality of the data, while maintaining the quality of the data. The final hypothesis states that evaluating the importance of features in heterogeneous data is possible and effective through the use of multiple unit models. The first chapters of the thesis address the basic issues related to proteomics and mass spectrometry. First, the general description of mass spectrometry and mass spectrometric imaging of biological samples is described. This is followed by a description of the main ionization methods and mass analyzers commonly used for the analysis of biological samples, especially samples from cancer patients. Then, there is a brief description of sample preparation, as well as data acquisition, its characteristics, and the initial steps taken to prepare the data for further analysis. These steps are baseline correction, normalization and alignment of the spectra.

The next chapter deals with the aggregation of mass spectra and the state of the art in peak detection. Peak detection was performed on the aggregated data using the most commonly used for this purpose methods. First, peaks were identified using a simple method based on the signal-to-noise ratio of peak intensities. Then peaks were identified with a peak modeling method based on the continuous wavelet transform.

In the following chapter, a more complicated method of peak identification was described in detail. With this method, peaks are identified by splitting the spectrum into smaller fragments and modeling them with Gaussian mixture models. First, a new

signal splitting method is described that differs from the method proposed in the original paper. A detailed operation scheme is described, and compared with the original method as well as the pseudocode for the algorithm implementation. The next section of the paper deals with the process of fitting the parts of the spectrum with Gaussian mixture models, with the general and mathematical description of the custom implementation of the expectation-maximization (EM) algorithm used for the fitting of Gaussian mixtures. The thesis also describes the selection of the optimal number of elements in the mixture and the influence of the stochastic nature of the EM algorithm on the results. All peak identification methods are compared to each other. The results of proposed peak identification method confirm the validity of the first hypothesis.

The sixth chapter describes the entire process of feature engineering. The deals with the use of statistics and spatial distribution to remove redundancy in the data and reduce the dimensionality of the data. To this end, noise was filtered using the parameters of the normal distributions that make up the spectrum model. Feature engineering is then continued by using the information provided by the imaging. The spatial distributions of nearby elements of the spectrum model are compared. The comparison is made using Peacock's statistical test for similarity of distributions. This statistical test is an extension of the Kolmogorov-Smirnov test to two dimensions. The critical values are calculated experimentally, and then the nearby elements with statistically identical special distribution are merged. The dimensionality reduction process ends with the detection of isotopic envelopes, which are also reduced to a single feature. Isotopic envelopes are detected by examining the distance between successive peaks, their shape, and their spatial distribution. The results show a significant reduction in the dimensionality of the data, from 9454 elements of the spectrum model to 888 features in the final set. These results confirm the second hypothesis of the paper.

The following sections describe the training of the classifiers on the processed data. Two groups of classifiers were trained. The first group was trained with an algorithm that uses multinomial logistic regression. The model is trained by iteratively performing logistic regression to find the best feature from the remaining set and adding it to the predictor list of the final model. The second set of classifiers are fully connected neural networks with two hidden layers, where the number of nodes is equal to the number of features. The performance of the classifiers was evaluated using metrics such as accuracy, precision, negative predictive value, sensitivity, specificity, f1 score, and ROC curves, precision-sensitivity curves, and their areas under the curve.

The process of feature importance evaluation was described next. Feature importance is assessed by assigning a score to each feature in unit models and averaging the results to determine the total feature importance score. For logistic regression models, scores are based on the feature's place in the predictor list. For neural networks, black-box model interpretation methods were used, LIME and Shapley values.

The last chapter of the thesis is the discussion about the experiments, the results, the conclusions drawn, and the goals for the future.

Streszczenie

Przedmiotem pracy doktorskiej jest analiza danych otrzymanych za pomocą obrazowania spektrometrią mas próbek pobranych od pacjentów z nowotworem głowy i szyi. W ramach pracy postawiono następujące hipotezy. Pierwsza hipoteza twierdzi, że identyfikacja pików w spektrach masowych może być skutecznie przeprowadzona za pomocą modelowania całego spektrum, poprzez podzielenie go na części oraz zamodelowaniu ich mieszaninami normalnymi. Druga hipoteza twierdzi, że informacja o przestrzennej dystrybucji danych pozyskana z obrazowania spektrometrią mass skutecznie usuwa redundancje i znacznie zmniejsza wymiarowość danych, przy jednoczesnym zachowaniu jakości danych. Ostatnia hipoteza twierdzi, że identyfikacja najważniejszych cech, dla danych heterogenicznych jest możliwa i skuteczna dzięki wnioskowaniu na podstawie wielu modeli jednostkowych.

Pierwsze rozdziały skupiają się na podstawowych zagadnieniach związanych z proteomiką i spektrometrią mas. W pierwszej kolejności przedstawiono ogólny opis spektrometrii mas oraz obrazowania tkanek za pomocą tej techniki. Opisane są najważniejsze metody jonizacji oraz analizatory mas, które są powszechnie wykorzystywane do analizy próbek pochodzenia biologicznego, w szczególności próbek pochodzących od pacjentów z nowotworem. Następnie po krótce opisano proces przygotowania próbek oraz pozyskiwania danych, ich charakterystykę, a także pierwsze działania mające na celu przygotowanie danych do dalszej analizy, tj. zastosowane metody korekty linii bazowej oraz normalizacji.

Następny rozdział porusza temat agregacji spektrów masowych oraz przedstawia aktualny wiedzy na temat detekcji pików. Na zagregowanych danych przeprowadzono detekcję pików przy pomocy najczęściej wykorzystywanych w tym celu algorytmów. W pierwszej kolejności piki zostały zidentyfikowane za pomocą najbardziej podstawowej i popularnej metody bazującej na określeniu współczynniku sygnału do szumu, wykorzystując intensywność pików jako sygnał. Następną wypróbowaną metodą jest, bazująca na transformacie falkowej, metoda modelowania pików.

W następnym rozdziale przedstawiona została metoda identyfikacji pików polegająca na modelowaniu całego spektrum masowego za pomocą modelu mieszanin rozkładów normalnych. Na początku opisana została metoda dzielenia spektrum masowego na mniejsze fragmenty w sposób odmienny od oryginalnie proponowanej. Przedstawiono szczegółowy schemat działania, wraz z pseudokodem oraz porównaniem wyników z oryginalną metodą.

Kolejna część pracy porusza temat dopasowywania mieszanin rozkładów normalnych do podzielonego spektrum. Zawiera ona matematyczny opis dopasowywania mieszanin z wykorzystaniem własnej implementacji algorytmu expectation-maximization (EM). W szczególności opisano metodę wybierania optymalnej liczby elementów w mieszaninach oraz wpływu losowej natury algorytmu EM na wyniki modelowania spektrum. Na końcu rozdziału przedstawiono wyniki procesu identyfikacji pików. Wyniki potwierdzają prawdziwość pierwszej postawionej hipotezy.

Następny rozdział skupia się na wykorzystywaniu metod statystycznych oraz przestrzennej dystrybucji cech w celu usunięcia redundancji i redukcji wymiarowości danych. W tym celu przeprowadzona została filtracja szumów wykorzystując parametry dystrybucji normalnych opisujących elementy modelu spektrum. Dalej inżynieria cech jest kontynuowana z wykorzystaniem informacji, których dostarcza obrazowanie. Przestrzenna dystrybucja pobliskich cech jest wykorzystana do zmniejszenia liczby cech. Porównanie jest wykonywane z pomocą testu statystycznego na podobieństwo dystrybucji Peacock'a. Jest to rozszerzenie testu Kołmogorov'a-Smirnov'a do dwóch wymiarów. Po wyznaczeniu eksperymentalnie wartości krytycznych, pobliskie cechy o statystycznie identycznej dystrybucji przestrzennej są łączone. Proces redukcji wymiarowości kończy się na detekcji obwiedni izotopowych, które są redukowane do pojedynczej cechy. Obwiednie izotopowe są wykrywane na podstawie odległości między pikami, kształtu pików oraz dystrybucji przestrzennej. Wyniki pokazują znaczne zmniejszenie wymiarowości danych, potwierdzając drugą postawioną hipotezę.

W kolejnych rozdziałach opisano proces uczenia klasyfikatorów na przetworzonych danych. Nauczono dwie grupy klasyfikatorów. Pierwsza grupa to klasyfikatory trenowane z wykorzystaniem wielomianowej regresji logistycznej, gdzie klasyfikator jest trenowany przez iteracyjne wykonywanie regresji logistycznej, wybierając za każdym razem najistotniejszą cechę ze zbioru i dodając ją do listy predyktorów końcowego modelu. Druga grupa klasyfikatorów to proste w pełni połączone sieci neuronowe z dwoma ukrytymi warstwami, każda z liczbą węzłów równą liczbie cech.

Klasyfikatory zostały ocenione między innymi za pomocą miar obliczanych na podstawie macierzy błędów takich jak dokładność, precyzja, czułość, swoistość, wartość predykcyjna ujemna, miara F1, a także krzywych ROC oraz krzywych dokładność-czułość. Ostatnia część eksperymentów bada słuszność trzeciej postawionej tezy. Jest ona poświęcona badaniom ogólnej ważności cech obu modeli, wykorzystując ważności cech w modeli jednostkowych. W szczególności, dla sieci neuronowych do określenia ważności cech wykorzystano takie metody jak LIME oraz wartości Shapley'a.

Ostatni rozdział pracy to dyskusja na temat przeprowadzonych badań, wniosków jakie zostały wyciągnięte podczas ich przeprowadzania, ich wyników oraz planów na dalsze badania.

List of Figures

List of Tables

Bibliography

[1] M. Vailati-Riboni, V. Palombo and J. J. Loor, “What Are Omics Sciences?,” *Periparturient Diseases of Dairy Cows: A Systems Biology Approach, Springer International Publishing*, pp. 1-7, 2017.

[2] R. Aebersold and M. Mann, "Mass spectrometry-based proteomics," *Nature 422,* pp. 198-207, 2003.

[3] B. Domon and R. Aebersold, "Mass spectrometry and protein analysis," *Science 312(5771),* pp. 212-217, 2006.

[4] A. R. Buchberger, K. DeLaney, J. Johnson and L. Li, "Mass Spectrometry Imaging: A Review of Emerging Advancements and Future Insights.," *Analytical chemistry 90(1),* pp. 240-265, 2018.

[5] R. M. Caprioli, T. B. Farmer and J. Gile, "Molecular imaging of biological samples: localization of peptides and proteins using MALDI-TOF MS.," *Analytical Chemistry 69(23),* pp. 4751-4760, 1997.

[6] J. Urban and D. Štys, "Noise and Baseline Filtration in Mass Spectrometry," in *Bioinformatics and Biomedical Engineering, Lecture Notes in Computer Sciences 9044,* Cham, Springer International Publishing, 2015, pp. 418-425.

[7] G. Wells, H. Prest i C. W. I. Russ, „Signal, Noise, and Detection Limits in Mass Spectrometry,” *Chemical Analysis, Application Note for Agilent Technologies,* 2023.

[8] H. Hu i J. Laskin, „Emerging Computational Methods in Mass Spectrometry,” *Advanced Science,* tom 9, 17 10 2022.

[9] E. F. Petricoin, A. M. Ardekani, B. A. Hitt, P. J. Levine, V. A. Fusaro, S. M. Steinberg, G. B. Mills, C. Simone, D. A. Fishman, E. C. Kohn and L. A. Liotta, "Use of proteomic patterns in serum to identify ovarian cancer," *The Lancet 359(9306),* pp. 572-577, 2002.

[10] G. Ball, S. Mian, F. Holding, R. O. Allibone, J. Lowe, S. Ali, G. Li, S. McCardle, I. O. Ellis, C. Creaser and R. C. Rees, "An integrated approach utilizing artificial neural networks and SELDI mass spectrometry for the classification of human tumours and rapid identification of potential biomarkers," *Bioinformatics 18(3),* pp. 395-404, 2002.

[11] R. H. Lillien, H. Farid and B. R. Donald, "Probabilistic disease classification of expression-dependent proteomic data from mass spectrometry of human serum," *Journal of Computational Biology 10(6),* pp. 925-946, 2003.

[12] Y. Li and Y. Liu, "A wrapper feature selection method based on simulated annealing algorithm for prostate protein mass spectrometry data," in *2008 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology*, Sun Valley, 2008.

[13] Y. Liu, "Feature extraction and dimensionality reduction for mass spectrometry data," *Computers in Biology and Medicine 39(9),* vol. 39, no. 9, pp. 818-823, 2009.

[14] G. Mrukwa and J. Polańska, "DiviK: divisive intelligent K-means for hands-free unsupervised clustering in big biological data," *BMC Bioinformatics 23(538),* no. 23, 2022.

[15] K. Bednarczyk, M. Gawin, M. Chekan, A. Kurczyk, G. Mrukwa, M. Pietrowska, J. Polańska and

P. Widłak, "Discrimination of normal oral mucosa from oral cancer by mass spectrometry imaging of proteins and lipids," *Journal of Molecular Histology 50(1),* pp. 1-10, 2019.

[16] R. J. Pais, R. Zmuidinaite, S. A. Butler and R. K. Iles, "An automated workflow for MALDI-ToF mass spectra pattern identification on large data sets: An application to detect aneuploidies from pregnancy urine," *Informatics in Medicine Unlocked 16(100194)*, vol. 16, 2019.

[17] H. Thiele, S. Heldmann, D. Trede, J. Strehlow, S. Wirtz, W. Dreher, J. Berger, J. Oetjen, J. H. Kobarg, B. Fischer i P. Maass, „2D and 3D MALDI-imaging: conceptual strategies for visualization and data mining," *Biochimica et Biophysica Acta,* pp. 117-137, Jan 2014.

[18] A. Polański, M. Marczyk, M. Pietrowska, P. Widłak and J. Polańska, "Signal partitioning algorithm for highly efficient gaussian mixture modeling in mass spectrometry," *PLoS One 10(7),* vol. 10, no. 7, 2015.

[19] S. S. Rubakhin i J. V. Sweedler, „A mass spectrometry primer for mass spectrometry imaging," *Methods in molecular biology (Clifton, N.J.),* pp. 21-49, 2010.

[20] G. L. Glish i R. W. Vachet, „The basics of mass spectrometry in the twenty-first century," *Nature Reviews Drug Discovery,* pp. 140-150, 01 Feb 2003.

[21] N. L. Anderson i N. G. Anderson, „Proteome and proteomics: new technologies, new concepts, and new words," *Electrophoresis,* tom 19, nr 11, pp. 1853-1861, Aug 1998.

[22] M. Wilm, „Principles of electrospray ionization.," *Molecular & cellular proteomics : MCP,* Jul 2011.

[23] G. I. Taylor, „Disintegration of water drops in an electric field," *Proc. R. Soc. Lond,* tom 280, p. 383–397, 28 Jul 1964.

[24] Z. Takáts, J. M. Wiseman, B. Gologan i R. G. Cooks, „Mass spectrometry sampling under ambient conditions with desorption electrospray ionization," *Science (New York, N.Y.),* tom 306, p. 471–473, 15 Oct 2004.

[25] K. Chughtai i R. M. A. Heeren, „Mass Spectrometric Imaging for Biomedical Tissue Analysis," *Chemical Reviews,* pp. 3237-3277, 28 Apr 2010.

[26] M. J. He, W. Pu, X. Wang, W. Zhang, D. Tang i Y. Dai, „Comparing DESI-MSI and MALDI-MSI Mediated Spatial Metabolomics and Their Applications in Cancer Studies," *Frontiers in Oncology,* 18 Jul 2022.

[27] P. J. Roach, J. Laskin i A. Laskin, „Nanospray desorption electrospray ionization: an ambient method for liquid-extraction surface sampling in mass spectrometry," *The Analyst,* tom 135, nr 9, p. 2233–2236, Sep 2010.

[28] Z. Luo, J. He, Y. Chen, J. He, T. Gong, F. Tang, X. Wang, R. Zhang, L. Huang, L. Zhang, H. Lv, S. Ma, Z. Fu, X. Chen, S. Yu i Z. Abliz, „Air flow assisted ionization for remote sampling of ambient," *Analytical Chemistry,* pp. 2977-2982, 5 Feb 2013.

[29] L. J. Gamble i C. R. Anderton, „Secondary Ion Mass Spectrometry Imaging of Tissues, Cells, and Microbial Systems.," *Microscopy today,* pp. 24-31, 18 Mar 2016.

[30] S. Yoon i T. G. Lee, „Biological tissue sample preparation for time-of-flight secondary ion mass spectrometry (ToF–SIMS) imaging," *Nano Convergence,* 26 Sep 2018.

[31] P. Nemes i A. Vertes, „Laser ablation electrospray ionization for atmospheric pressure molecular imaging mass spectrometry.," *Methods in molecular biology (Clifton, N.J.),* p. 159–171, 2010.

[32] S. M. Khalil, A. Römpp, J. Pretzel, K. Becker i B. Spengler, „Phospholipid Topography of Whole-Body Sections of the Anopheles stephensi Mosquito, Characterized by High-Resolution Atmospheric-Pressure Scanning Microprobe Matrix-Assisted Laser Desorption/Ionization Mass Spectrometry Imaging," *Analytical chemistry,* p. 11309–11316, 17 Nov 2015.

[33] M. Ekelöf, E. K. McMutrie, M. Nazari, S. D. Johanningsmeier i D. C. Muddiman, „Direct Analysis of Triterpenes from High-Salt Fermented Cucumbers Using Infrared Matrix-Assisted Laser Desorption Electrospray Ionization (IR-MALDESI).," *Journal of the American Society for Mass Spectrometry,* p. 370–375, Feb 2017.

[34] N. T. N. Phan, A. S. Mohammadi, M. D. Pour i A. G. Ewing, „Laser Desorption Ionization Mass

Spectrometry Imaging of Drosophila Brain Using Matrix Sublimation versus Modification with Nanoparticles," *Analytical chemistry,* p. 1734–1741, 02 Feb 2016.

[35] R. Haddad, H. M. S. Milagre, R. R. Catharino i M. N. Eberlin, „Easy Ambient Sonic-Spray Ionization Mass Spectrometry Combined with Thin-Layer Chromatography," *Analytical Chemistry,* p. 2744–2750, 11 Mar 2008.

[36] W. E. Stephens, B. Serin i W. E. Meyerhof, „A Method for Measuring Effective Contact e.m.f. between a Metal and a Semi-conductor," *Physical Review Journals,* tom 69, 01 Jan 1946.

[37] D. Price, „The Early Years as Chronicled by the European Time-of-Flight Symposia," w *Time-of-flight mass spectrometry,* American Chemical Society, 1994, pp. 1-15.

[38] A. E. Cameron i D. F. Eggers, „An Ion "Velocitron"," *Review of Scientific Instruments,* tom 19, nr 9, 16 Jun 1948.

[39] W. C. Wiley i I. H. McLaren, „Time-of-Flight Mass Spectrometer with Improved Resolution," *Review of Scientific Instruments,* p. 1150, Jan 1955.

[40] A. E. Clark, E. J. Kaleta, A. Arora i D. M. Wolk, „Matrix-assisted laser desorption ionization-time of flight mass spectrometry: a fundamental shift in the routine practice of clinical microbiology," *Clinical microbiology reviews,* pp. 547-603, Jul 2013.

[41] D. S. Cornett, S. L. Frappier i R. M. Caprioli, „MALDI-FTICR imaging mass spectrometry of drugs and metabolites in tissue," *Analytical Chemistry,* tom 80, nr 14, pp. 5648-53, 15 Jul 2008.

[42] N. Singhal, M. Kumar, P. K. Kanaujia i J. S. Birdi, „MALDI-TOF mass spectrometry: an emerging technology for microbial identification and diagnosis," *Frontiers in Microbiology,* 05 Aug 2015.

[43] A. C. Crecelius, D. S. Cornett, R. M. Caprioli, B. Williams, B. M. Dawant i B. Bodenheimer, „Three-Dimensional Visualization of Protein Expression in Mouse Brain Structures Using Imaging Mass Spectrometry," *Journal of the American Society for Mass Spectrometry,* tom 16, nr 7, pp. 1093-1099, Jul 2005.

[44] E. H. Seeley i R. M. Caprioli, „3D Imaging by Mass Spectrometry: A New Frontier," *Analytical Chemistry,* pp. 2105-2110, 24 Jan 2012.

[45] N. Jeffreis, „Algorithms for alignment of mass spectrometry proteomic data," *Bioinformatics,* tom 21, p. 3066–3073, Jul 2005.

[46] H. López-Fernández, H. M. Santos, J. L. Capelo, F. Fdez-Riverola, D. Glez-Peña i M. Jato-Reboiro, „Mass-Up: an all-in-one open software application for MALDI-TOF mass spectrometry knowledge discovery," *BMC Bioinformatics,* tom 16, 2015.

[47] S. Gibb i K. Strimmer, „MALDIquant: a versatile R package for the analysis of mass spectrometry data," *Bioinformatics,* pp. 2270-2271, 01 Sep 2012.

[48] P. Du, W. A. Kibbe i S. M. Lin, „Improved peak detection in mass spectrum by incorporating continuous wavelet transform-based pattern matching.," *Bioinformatics (Oxford, England),* pp. 2059-2065, 01 Sep 2006.

[49] J. W. Wong, G. Cagney i H. M. Cartwright, „SpecAlign—processing and alignment of mass spectra datasets," *Bioinformatics,* tom 21, nr 9, p. 2088–2090, 09 May 2005.

[50] D. May, W. Law, M. Fitzgibbon, Q. Fang i M. McIntosh, „A software platform for rapidly creating computational tools for mass spectrometry-based proteomics," *Journal of proteome research,* tom 8, nr 6, p. 3212–3217, 2009.

[51] M. Strum, A. Bertsch, C. Gröpl, A. Hildebrandt, R. Hussong, E. Lange, N. Pfeifer, O. Schulz-Trieglaff, A. Zerck, K. Reinert i O. Kohlbacher, „OpenMS – An open-source software framework for mass spectrometry," *BMC Bioinformatics,* 26 Mar 2008.

[52] J. Kolibal i D. Howard, „MALDI-TOF baseline drift removal using stochastic Bernstein approximation," 01 12 2006.

[53] H. Shin, M. P. Sampat, J. M. Koomen i M. K. Markey, „Wavelet-based adaptive denoising and baseline correction for MALDI TOF MS," *Omics : a journal of integrative biology,* pp. 283-295, Jun 2010.

[54] K. Bednarczyk, M. Gawin, M. Pietrowska, P. Widłak i J. Polańska, „Adaptive baseline correction algorithm for MALDI spectra," Doorn, The Netherlands, 2017.

[55] C. Bruffaerts, V. Verardi i C. Vermandele, „A generalized boxplot for skewed and heavy-tailed distributions," *Statistics & Probability Letters,* pp. 110-117, Dec 2014.

[56] J. W. Wong, C. Durante i H. M. Cartwright, „Application of fast Fourier transform cross-correlation for the alignment of large chromatographic and spectral datasets," *Analytical chemistry,* p. 5655–5661, 2005.

[57] J. S. Morris, K. R. Coombes, J. Koomen i K. A. Baggerly, „Feature extraction and quantification for mass spectrometry in biomedical applications using the mean spectrum," *Bioinformatics (Oxford, England),* pp. 1764-75, Jun 2005.

[58] C. Yang, Z. He and W. Yu, "Comparison of public peak detection algorithms for MALDI mass spectrometry data analysis," *BMC Bioinformatics,* 06 Jan 2009.

[59] C. Bauer, R. Cramer i J. Schuchhardt, „Evaluation of peak-picking algorithms for protein mass spectrometry.," *Methods in molecular biology (Clifton, N.J.),* pp. 341-352, 2011.

[60] H. Zhou, „Signal-to-Noise (SNR) and Uncertainty Estimates," 2019. [Online]. Available: https://nmr.chem.ucsb.edu/protocols/SNR.html. [Data uzyskania dostępu: 23 Jan 2022].

[61] I. Daubechies, Ten Lectures on Wavelets, Society for Industrial and Applied Mathematics, 1992.

[62] J. Polańska, M. Plechawska, M. Pietrowska i Ł. Marczyk, „Gaussian mixture decomposition in the analysis of MALDI-TOF spectra," *Expert Systems 29(3),* pp. 216-231, 09 Mar 2011.

[63] M. Plechawska-Wójcik, „Mathematical Model of Mass Spectrometry Data Based on Gaussian Mixture Models," *Advanced Science Letters,* Feb 2014.

[64] R. H. C. Lopes, „Kolmogorov-Smirnov Test," w *International Encyclopedia of Statistical Science,* Springer Berlin Heidelberg, 2011, pp. 718-720.

[65] J. A. Peacock, „Two-dimensional goodness-of-fit testing in astronomy," *Monthly Notices of the Royal Astronomical Society,* tom 202, nr 3, p. 615–627, Mar 1983.

[66] R. A. Fisher i M. A, Statistical Methods for Research Workers, Edinburgh: Oliver and Boyd, 9251.

[67] F. Dematheis, M. C. Walter, D. Lang, M. Antwerpen, H. C. Scholz, M.-T. Pfalzgraf, E. Mantel, C. Hinz, R. Wölfel i S. Zange, „Machine Learning Algorithms for Classification of MALDI-TOF MS Spectra from Phylogenetically Closely Related Species Brucella melitensis, Brucella abortus and Brucella suis," *Microorganisms,* 17 Aug 2022.

[68] P. Lasch, W. Beyer, H. Nettermann, M. Stammler, E. Siegbrecht, R. Grunow i D. Naumann, „Identification of Bacillus anthracis by Using Matrix-Assisted Laser Desorption Ionization-Time of Flight Mass Spectrometry and Artificial Neural Networks," *Applied and environmental microbiology,* tom 75, nr 22, p. 7229–7242, 2009.

[69] C. Gonzales, X. A. López-Cortés i S. Maldonado, „Semi-supervised learning for MALDI–TOF mass spectrometry data classification: an application in the salmon industry," *Neural Computing and Applications,* 21 feb 2023.

[70] T. Motier, A. Wieme, P. Vandamme i W. Waegeman, „Bacterial species identification using MALDI-TOF mass spectrometry and machine learning techniques: A large-scale benchmarking study," *Journal, Computational and Structural Biotechnology,* tom 19, nr 10, Nov 2021.

[71] M. H. Zweig i G. Campbell, „Receiver-operating characteristic (ROC) plots: a fundamental evaluation tool in clinical medicine," *Clinical Chemistry,* tom 39, pp. 561-577, 01 Apr 1993.

[72] R. Roscher, B. Bohn, M. F. Duarte i J. Gracke, „Explainable Machine Learning for Scientific Insights and Discoveries," *IEEE Access,* tom 8, pp. 42200-42216, 2020.

[73] P. Linardatos, V. Papastefanopoulos i S. Kotsiantis, „Explainable AI: A Review of Machine Learning Interpretability Methods," *Entropy,* 25 Dec 2020.

[74] P. Angelov i E. Soares, „Towards explainable deep neural networks (xDNN)," *Neural Networks,* pp. 185-194, Oct 2020.

[75] C. Molnar, Interpretable Machine Learning: A Guide for Making Black Box Models

Explainable, 2 red., 2022.

[76] M. T. Ribeiro, S. Sameer i G. Carlos, „Why should I trust you?: Explaining the predictions of any classifier.," w *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 2016.

[77] L. S. Shapley, „A value for n-person games.," *Contributions to the Theory of Games*, pp. 307-317, 21 Mar 1953.

[78] E. Winter, „Chapter 53 The shapley value," w *Handbook of Game Theory with Economic Applications*, 2002, pp. 2025-2054.