FAKULTÄT FÜR !NFORMATIK
Faculty of Informatics

# Webbasierte Visualisierung der Klassifizierung und Community-Erkennung in medizinischen Daten

BACHELORARBEIT

zur Erlangung des akademischen Grades

**Bachelor of Science**

im Rahmen des Studiums

**Software & Information Engineering**

eingereicht von

**Aleksandar Djuric**
Matrikelnummer 01227873

an der Fakultät für Informatik
der Technischen Universität Wien
Betreuung: Univ. Ass. Renata Raidou, MSc PhD

Wien, 11. März 2020

| Aleksandar Djuric | Renata Raidou |
|---|---|

FAKULTÄT FÜR !NFORMATIK
Faculty of Informatics

# Web-based visualization of classification and community detection in medical data

BACHELOR'S THESIS

submitted in partial fulfillment of the requirements for the degree of

**Bachelor of Science**

in

**Software & Information Engineering**

by

**Aleksandar Djuric**
Registration Number 01227873

to the Faculty of Informatics
at the TU Wien
Advisor: Univ. Ass. Renata Raidou, MSc PhD

Vienna, 11th March, 2020

| Aleksandar Djuric | Renata Raidou |
|---|---|

# Erklärung zur Verfassung der Arbeit

Aleksandar Djuric

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wien, 11. März 2020

Aleksandar Djuric

# Danksagung

Ich möchte mich bei meiner Familie für ihre ununterbrochene Unterstützung danken. Ihre Standhaftigkeit und der damit verbundene Erfolg, ermöglichte es mir immer ein klares Ziel vor Augen zu haben und dieses mit voller Motivation zu verfolgen.

Im weiteren danke ich meinen Freunden, die mich mit Ablenkungen und Aufmunterungen auch in den schwierigsten Zeiten erheiterten und mir wieder Energie gaben.

Letzltlich danke ich meiner Betreuerin, für eine ausgezeichnete Betreuung während des gesamten Prozesses und des konstanten Inputs um das beste Ergebnis zu erzielen.

# Acknowledgements

I want to thank my family for their continuous support. Their success and perseverance leading to their achievements, made it possible for me to always have a clear goal and pursue my dream with full motivation.

Furthermore, I thank my friends for always keeping up my energy even in the darkest times, by distracting and encouraging me.

Last but not least I thank my advisor, for the excellent supervision throughout the whole process and constantly giving me great input to create the best result possible.

# Kurzfassung

Heutzutage stellt der Brustkrebs eine der meistverbreiteten Krebsarten bei der Frauenpopulation dar. Um den mit Brustkrebs kämpfenden Frauen zu helfen und die Tonanzahl unter den erkrankten Frauen zu reduzieren, ist es notwendig, die Symptome so früh wie möglich zu erkennen. Um das zu ermöglichen, analysieren klinische Erforscher die Symptome von erkrankten Frauen, um verstehen zu können, wie sich individuelle PatientInnen während der Behandlungen durch die Verwendung der Klassifikationen und anderer klinischen Klassifizierungen benehmen. Um die Analysen von Klassifizierungsprozessen zu begünstigen, wurde eine Web-Applikation für die Unterstützung visueller Analyse des Brustkrebs erstellt.

In dieser Arbeit wurden drei verschiedene Möglichkeiten für visuelle Darstellung der Daten des Brustkrebses diskutiert, welche in unserer Web-Applikation integriert wurden. Wir implementierten die Funktionalität, eine oder mehrere PatientInnen durch verschiedene Sichtweisen zu vergleichen, um mehrdimensionale Daten zu ermöglichen. Wir berücksichtigten drei Gruppen der Klinische Referenzdaten (Input der Klassifikatoren, Output der Klassifikatoren und klinisch nutzbare Daten für die Einteilung), welche in drei verschiedenen visuellen Repräsentationen dargestellt werden und miteinander verbunden werden. Auf diesen visuell interaktiven Repräsentationen basierend, können die ErforscherInnen, die an der automatisierten Klassifizierung der an Brustkrebs erkrankten Patientinnen arbeiten, diese Daten verstehen und analysieren. Durch ein paar Anwendungsfälle werden die Ergebnisse, der Verwendung dieser Web-Applikation entnommen, dargestellt. Die Ergebnisse zeigen visuell die Funktionalität, Vorteile und Möglichkeiten, die durch dieses Konzept unterstützt werden. Durch die Verwendung dieser Web-Applikation können die ErforscherInnen visuell recherchieren und die Daten eines Patienten/einer Patientin oder einer Gruppe verschiedener PatientInnen durch unterschiedliche Sichtweisen vergleichen, was vorher mithlfe anderer Forschungswerkzeuge nicht durchführbar war.

# Abstract

Breast cancer is one of the most common cancers in the female population. In order to help women with breast cancer and to reduce the mortality among affected women, it is necessary to detect the disease as soon as possible. To enable this, clinical researchers analyze the data they have from diseased women, in order to understand how individual groups of patients behave during a particular treatment with the use of classifiers and other clinical classification systems. To support the analysis of the classification process, we have created a web-based application that supports visual analysis of breast cancer data.

In this thesis, we will discuss three different possibilities to visually represent breast cancer data, which have been implemented in our web-based framework. We implemented functionality to compare patients through different views to enable easy exploration and analysis of the available multivariate data. We also consider three groups of data(input of the classifier, the output of the classifier and clinical reference data for classification) which are represented in three different visual representations that are linked to each other. Based on these interactive visual representations, researchers working on the automated classification of breast cancer patients can understand and analyze their data and their classifiers. Through several use cases, we demonstrate the results obtained by using this web-based framework. The results visually illustrate the functionality, benefits, and possibilities that this framework supports. Using this framework, researchers can visually explore and analyze the data of a patient or a group of different patients and compare it through three different views, which was not possible before with other exploratory tools.

# Contents

CHAPTER 1

# Introduction

## 1.1 Motivation

In this work, we want to develop an easy way for analyzing breast tumor classification outcomes for researchers working on classification algorithms. In the proposed implementation, we use multiple coordinated views and different interaction techniques as a means to facilitate analysis. For this application, we rely on brushing and linking interaction, to support and to understand what comes out from different classifiers and to compare them to different (clinical) classification schemes.

## 1.2 Goal of the Thesis

The goal of this thesis is to visually present patient data acquired in breast cancer research in the most visually effective way and to enable clinical researchers working on breast cancer classification to analyze their data and the outcomes of their classification algorithms. Through the adequate visual representation of the tumor data, we want to simplify and support the analysis of classification algorithm outcomes. Additionally, users can compare different data types from testing and training processes together with different clinical classification schemes, considering that they have several different visual representations at their disposal. We will answer the following questions:

1) How can we compare different types of tumors classifications employing several data representations?

2) What is the relationship between the input of the tumor tissue classifier and the output of the classifier?

3) What is the connection between the tumor classification and the different clinical classification schemes?

Our visualization makes it possible to compare different classifications of tumors and show the connection between input patient data features and output patient data classification outcomes through charts. It also enables the identification of certain points of interest and a more detailed analysis of the classification process, with regards to clinical classification. By using this representation the user, i.e. researchers developing tumor tissue classification algorithms, will find it possible to easily confirm some expected results and to conduct a detailed analysis for new hypothesis generation or confirmation.

## 1.3 Contribution

We developed a web-based interface to support developers of classification algorithms to analyze the outcome of breast tumor classification. Our application supports feature analysis and classification scheme comparison through multiple coordinated views. Through our web interface, we represent different showcases of breast tumor classification.

CHAPTER 2

# Related Work

In this work, we represent breast cancer through different charts and views. There are other ways in which multidimensional information can be presented and that will be discussed in this chapter. We will first introduce and explain some other works and their representations and then we will discuss differences, advantages, and disadvantages between our work and other works.

The WEAVE System [8] allows its users the creation of a wide range of possible visualizations. An example is shown in figure 2.1. With this system, the user can easily create a 3D view and compare it with more than 20 different multidimensional views, which are all automatically linked with one another. Multidimensional representations can be common plots such as histograms, pie charts and scatter plots to sophisticated presentations representing correlations between sets of variables, clustering of cases, and views of very high dimensionality data. This system presents an improvement in the visualization of scientific data. Users can easily and quickly analyze and compare data from different views, providing several viewpoints on the data. In my case, I co-analyze the input and output of the classifier with clinical classification schemes.

Another well-known system for visual analysis and interactive exploration and of various multi-dimensional and time-varying data is SimVis [5, 6, 20] (This is shown in figure 2.2). This is developed by the VRVis Research Center (Vienna, Austria), and uses such techniques to provide access to the data interactively and to explore and analyze large three-dimensional time-dependent fields. SimVis is a software tool that also enables users interactive experience through brushing and linking. In most cases is used to reflect climate changes. The ultimate goal of this system is to identify parameters and regions reacting most sensitive to climate change, representing robust indicators. This system displays most of its data through scatter plots while we have four different plots to represent our data. The difference between this system and our applications is that we do not have the anatomical view and we care more about the visualization of input/output data.

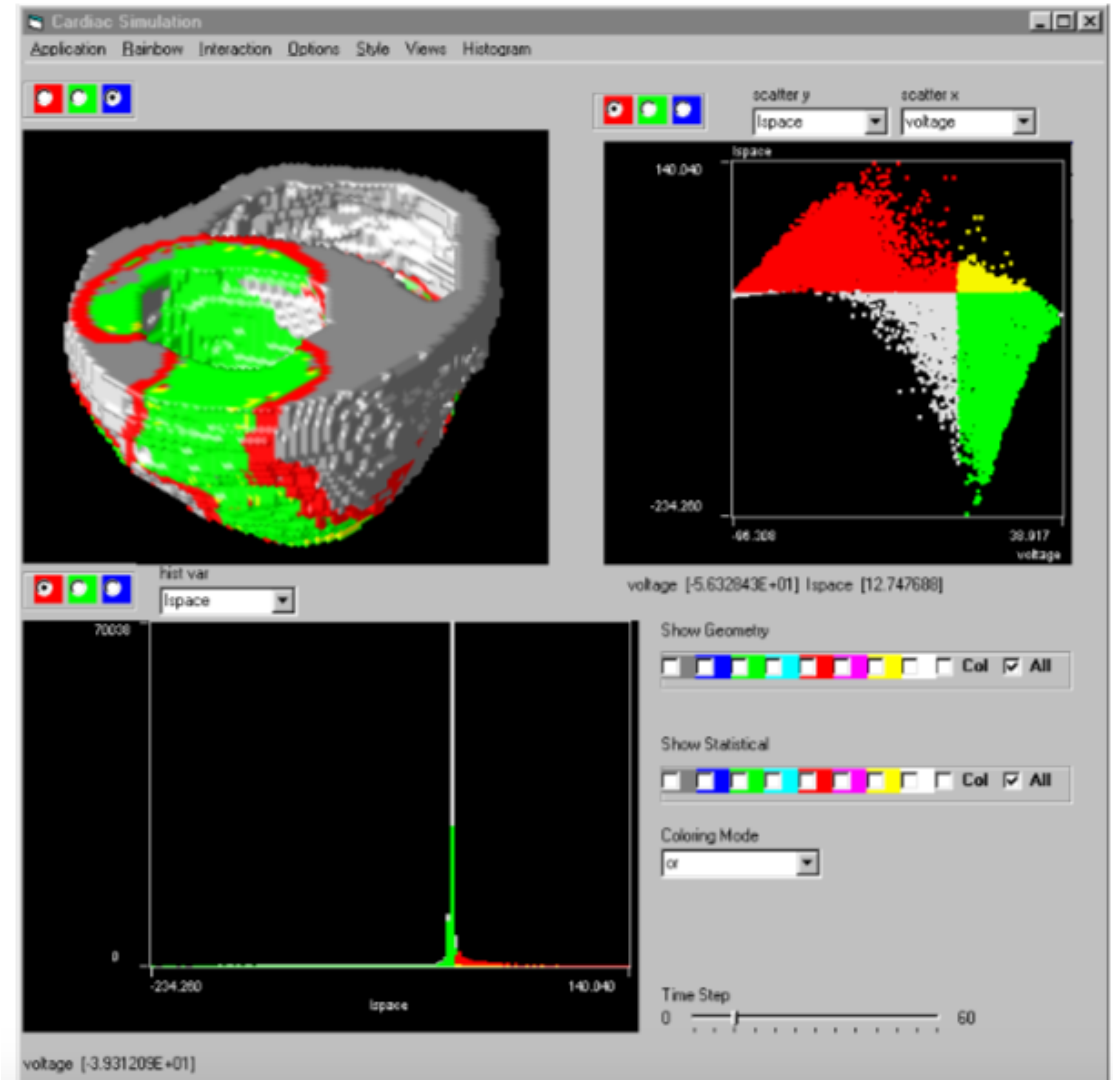



Figure 2.1: *WEAVE system example for the exploration of multidimesional data [8].*

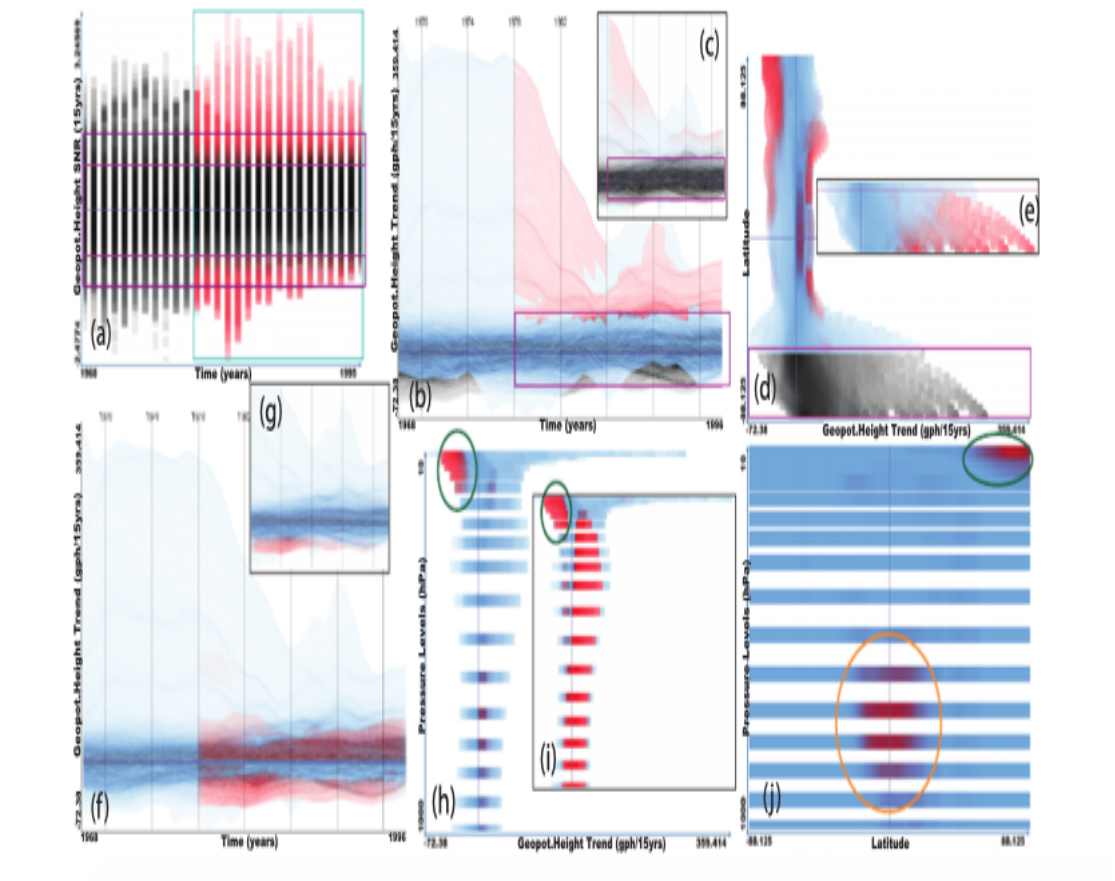



Figure 2.2: *SimVis system example for the visual analysis and interactive exploration of various multi-dimensional and time-varying data [5, 6, 20].*

Another example of interactive visual linear data projection is the work of Oeltze et al. This work allows users to explore perfusion data. Perfusion data is a dynamic medical image data that characterizes the regional blood flow in human tissue. In this work, researchers can display correlations and relations between different parameters and features of a given data. They also use time-intensity curves that characterize the amount of contrast agent enhancement of the perfusion data. Based on these curves, they can obtain the parameters used to diagnose the tumor. To reduce the dimensions of their parameter space they use Principle Component Analysis (PCA). PCA is a technique used to emphasize variation and bring out strong patterns in a dataset. The results are represented through multidimensional views, which are linked and allow users to easily, efficiently and quickly explore complex data. The users can also see some results in the 3D view. Their visualization of data is quite different from ours and is based on medical images, while we rely more on extracted features. An example of the work of Oeltze is shown in figure 2.3.

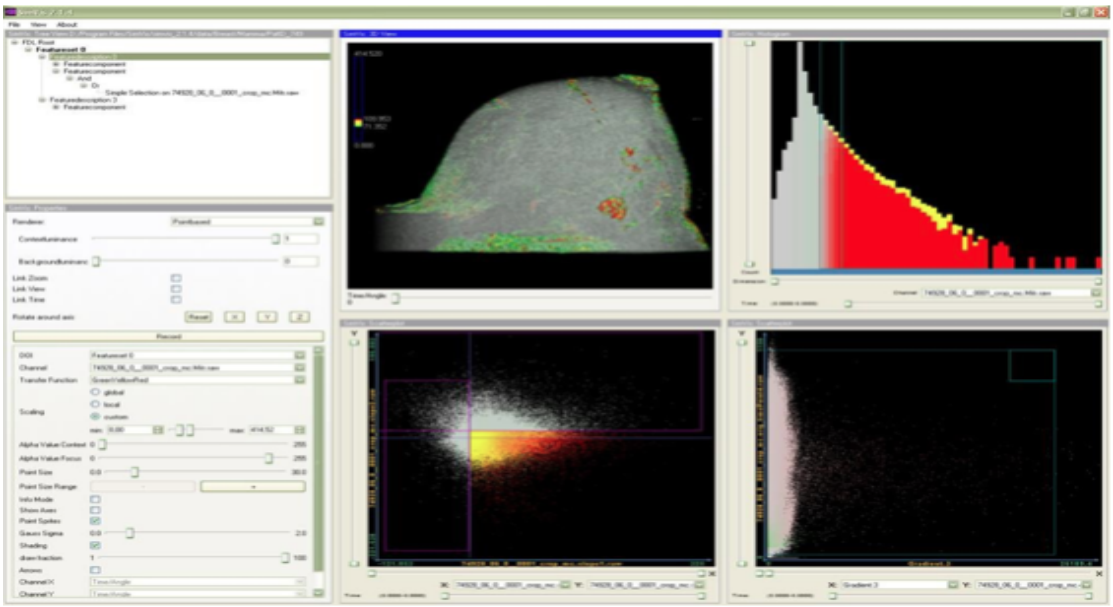


Figure 2.3: *Work of Oeltze et al. for the exploration of breast perfusion data [22].*

CHAPTER 3

# Methods

In this chapter, we analyze different methods that could be used to represent breast cancer classification and its outcome and we discuss their advantages and disadvantages. Also, we explain the reasons why we used certain methods to represent patient data, as opposed to alternative methods we could have used. We discuss the limits that our selected representations have and explain why we had to use the alternatives to obtain certain information/insights in data. The way the charts are linked with each other is also explained, and, lastly, the use of encodings in the representation is discussed.

Our data were sorted into three groups :

1) Data-input of classifier (high-D space of features)

2) Data-output of classifier (3-D space)

3) Multi-class data (different clinical classification schemes vs. output class of classifier).

## 3.1 Data Processing

We have been provided with a data table consisting of 22 columns and 180 rows obtained from a breast cancer study at Liverpool University. The rows are different patients of the study. The columns relate to the classification process. The first column is the *patient_id*, through which we can identify each patient, and the rest of the data is divided into several groups, based on which we have implemented our graphical visualizations. Values from the first 14 columns are classifier input features (*calc_median, calc_max, calc_variance, calc_skewness, glcm_inverseVariance_0, glcm_contrast_0, glcm_correlation_45, glcm_homogeneity2_0, glcm_IDN,glrlm_RLN_0, glrlm_SRHGLE, glrlm_LRHGLE_90, glrlm_GLN_90, glrlm_LRLGLE_90*). All these values are continuous numerical data which are features derived from the images for each patient. We do not have more detailed information on the feature extraction or the physical meaning of

the features. Also we do not have access to the medical data Based on this information, we set off to create parallel coordinates, as we will discuss later.

The next group consists of the output of the classifier which was acquired based on the data we have from the input of the classifier. These data consist of three values *x,y,z* which are also continuous numerical data. The output of the classifier is represented in a combination of a 3D scatter plot and a scatter plot matrix, as we will discuss later.

After that, we have clinical reference data for classification which consists of the following four columns (*tum_type, Cluster label, birads* and *subtlety*). The most important column about this group is tum_type showing the type of tumor for a particular patient. **Nor** represents normal tumor, **B** Benign, **M** Malignant and **L** Lesion tumor. Each tumor in our application is presented in a different color, which the user can customize. The *cluster labels* have discrete values between 1 and 6 and represent clusters of patients with similar characteristics resulting from clustering methods used by the developers. *Birads* stands for Breast Imaging Reporting and Data System [16, 23, 21]. This is a categorization used in clinical practice and is conducted by an expert radiologist. It has discrete values between 1 and 5. *Subtlety* represents a measure of how difficult/easy a case is. It is a numerical value from 1 to 33 [15].

The last group consists of one value, namely whether the data from the real testing process or from the training process. Training data are data that were used to train the algorithm. Test data are data that are used in the testing phase. If the value in this column is 1 then the data is from the training process and if it is 0, then the data is from the testing process. The data file we used to test the application consists of data for 180 patients. Half of this data is training and half is testing data.

## 3.2 Visualization for the input of the classifier

For the input of the classifier, we have 14 data features which were a result of different calculations and measurements on the medical images of the patients. By researching different ways of representing multidimensional data, we decided that the best way for these data to be shown is with parallel coordinates [14]. An example is shown in figure 3.1. One of the alternative solutions would be to use a scatter plot matrix but in this case, we would get a 14x14 matrix that would be very demanding in screen space and difficult for analysis. Parallel coordinates are a common way of visualizing and analyzing high-dimensional multivariate data. They are suitable for comparing many variables together and for analyzing the relationships between them. In parallel coordinates, each variable is assigned one axis and all the axes are placed in parallel to each other. Each axis can have a different scale, as each variable works off a different unit of measurement [11, 12].

One of the reasons why we decided to use parallel coordinates is that the user must have the ability to compare all 14 dimensions simultaneously in a 2D visualization. Parallel coordinates are particularly good for this, because we can easily represent, analyze and

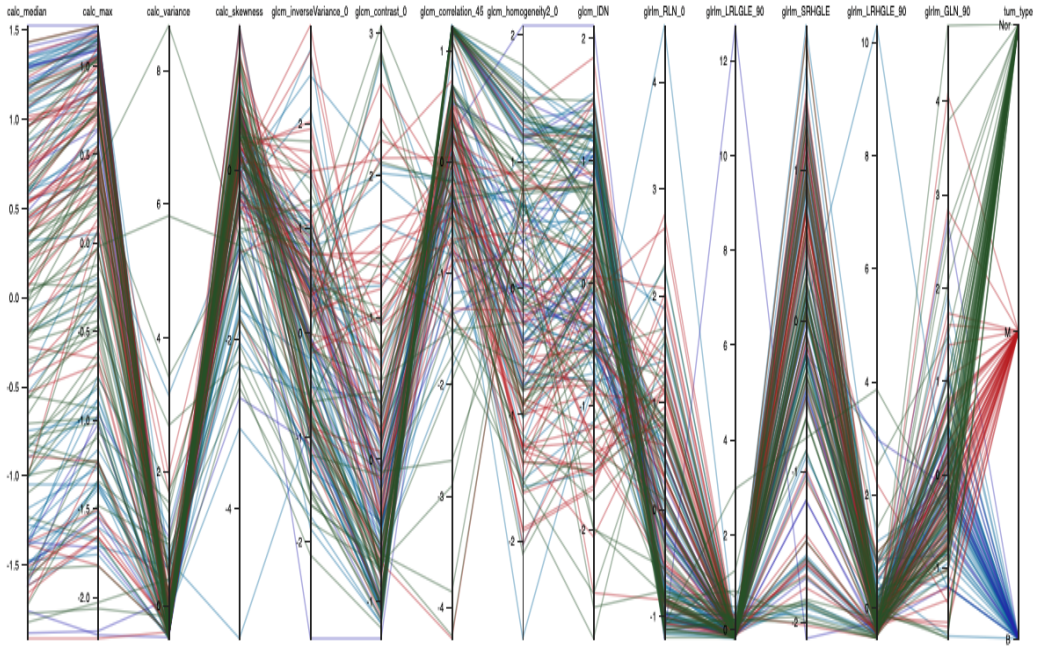



Figure 3.1: *Parallel coordinates showing the data of the input of the classifier.*

compare many dimensions. By adding additional functionality in parallel coordinates like *brushing* and *linking* we make data analysing even easier. Each patient is represent by a line passing through every vertical axis. Based on the intersection of the line with the axis we can see this value of a specific feature. This method enables us to show different data types with different colors. In this case, each tumor type is highlighted with a different color. The default color for benign tumor is blue, for malignant red, for normal green and for lesion purple. Tumor type colors are the same in each representation. For example, if the lines or points in representation are green, that is mean that this patient has a normal tumor. If we had a larger number of axes, there could be problems with displaying and analyzing the data. In our case having more than the 14 given axes, we would not have any problems during the creation of parallel coordinates in a technical sense, but we would have a problem with understanding and analyzing the data. Overplotting (overlaying data lines) is addressed by lowering the opacity of the lines. Reordering is also possible for an easier comparison of the data (This is shown in figure 3.2).

Another reason why we opted for this method is that it supports brushing, multi-brushing and reordering. Brushing enables us to easily select a part of the data based on the values of one feature that we find interesting, and in that way, we can analyze all different axes (This is shown in figure 3.3). Multi-brushing enables us to select certain data parts based on the values of multiple features through different axes at the same time and

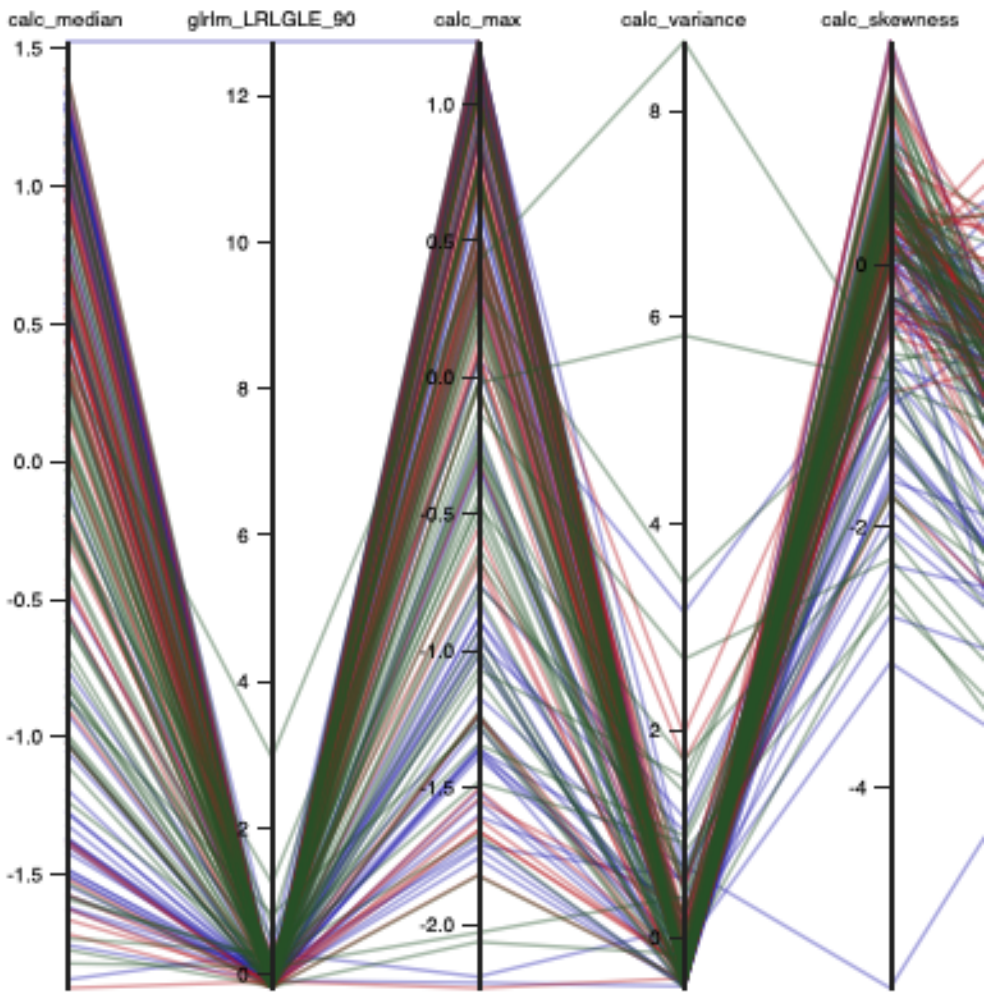
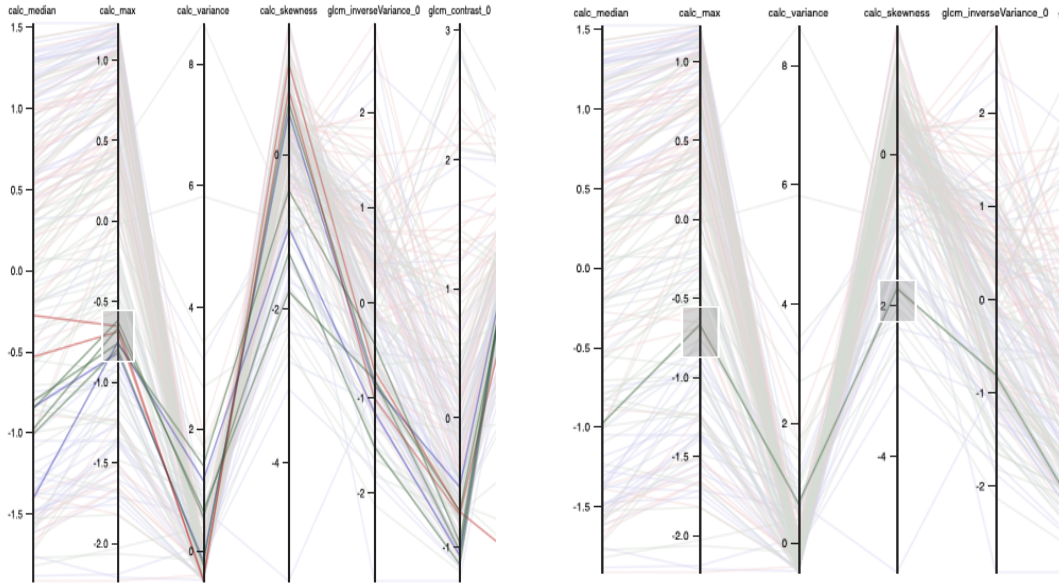



Figure 3.2: *Scaling example in parallel coordinates with different scale for each axes.*

analyze only the selected data (This is shown in figure 3.3). One disadvantage of parallel coordinates is that we cannot directly compare two axes unless they are placed right next to each other.The standard functionality of parallel coordinates enables us to change the placement of selected axes to compare and analyze data between specific dimensions. Also, parallel coordinates give us the ability to distribute the data into groups by bundling [9, 14]. With bundling, we can choose along which axis we want to conduct bundling, as well as bundling strength and curve smoothness to increase visibility. An example is shown in figure 3.4.




(a) One axis selected (Brushing) (b) Two axes selected (Multi-brushng)

Figure 3.3: *Brushing vs Multi-brushng.*

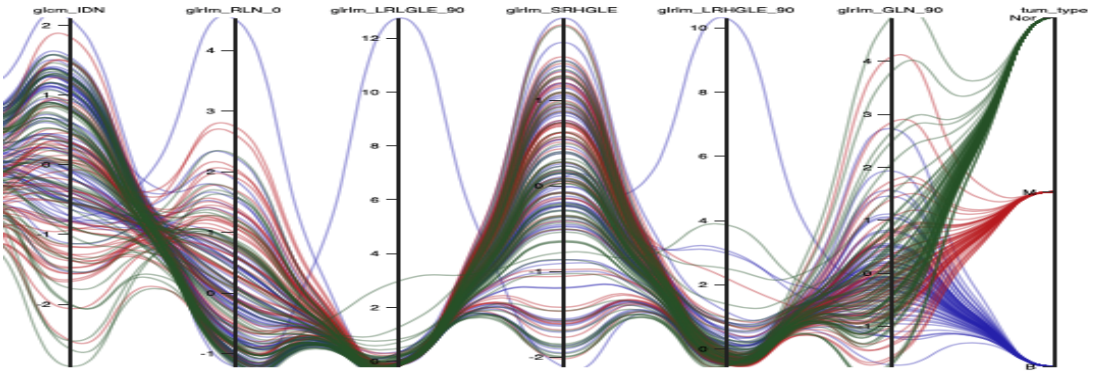



Figure 3.4: *Parallel coordinates bundling applied to input data which give us better view of some groups of data.*

## 3.3 Visualization of the output of the classifier

The output of the classifier is given to as a 3D space $(x,y,z)$, i.e. for each patient 14D $\rightarrow (x,y,z)$. As opposed to the 14 values we have for the input, the number of data for the output is significantly lower, which presents us with the possibility to show it in multiple ways. In this case, we could have used parallel coordinates again, but we are specifically interested in the position of certain values along with their layout and the relationship between certain patients and certain types of tumors. We are interested

in seeing proximities in the low dimensional space. That is why we decided to use a combination of 2D and 3D scatter plots to represent the output of the classifier. In order to show three different values through the 2D scatter plot, we had to distribute the data into three groups, so we could do pairwise comparisons (*xy,xz,yz*). These groups are parts of the scatter plot matrix [3, 25]. A n-D scatter plot matrix is a **n x n** matrix of 2D scatter plots (This is shown in figure 3.5). Each scatter plot in the matrix represents the relationship between two variables. We only used three scatter plot from matrix because the diagonal scatter plots comparing a dimension with itself and other scatter plots are same as our but just inverted. By using the 2D scatter plot matrix, we can see clearly how the data are grouped with different tumor types, and which data do not match with the predicted value. An example is shown in figure 3.6.


Figure 3.5: *Example of visualizing the four dimensions using a scatter plot matrix. [13].*

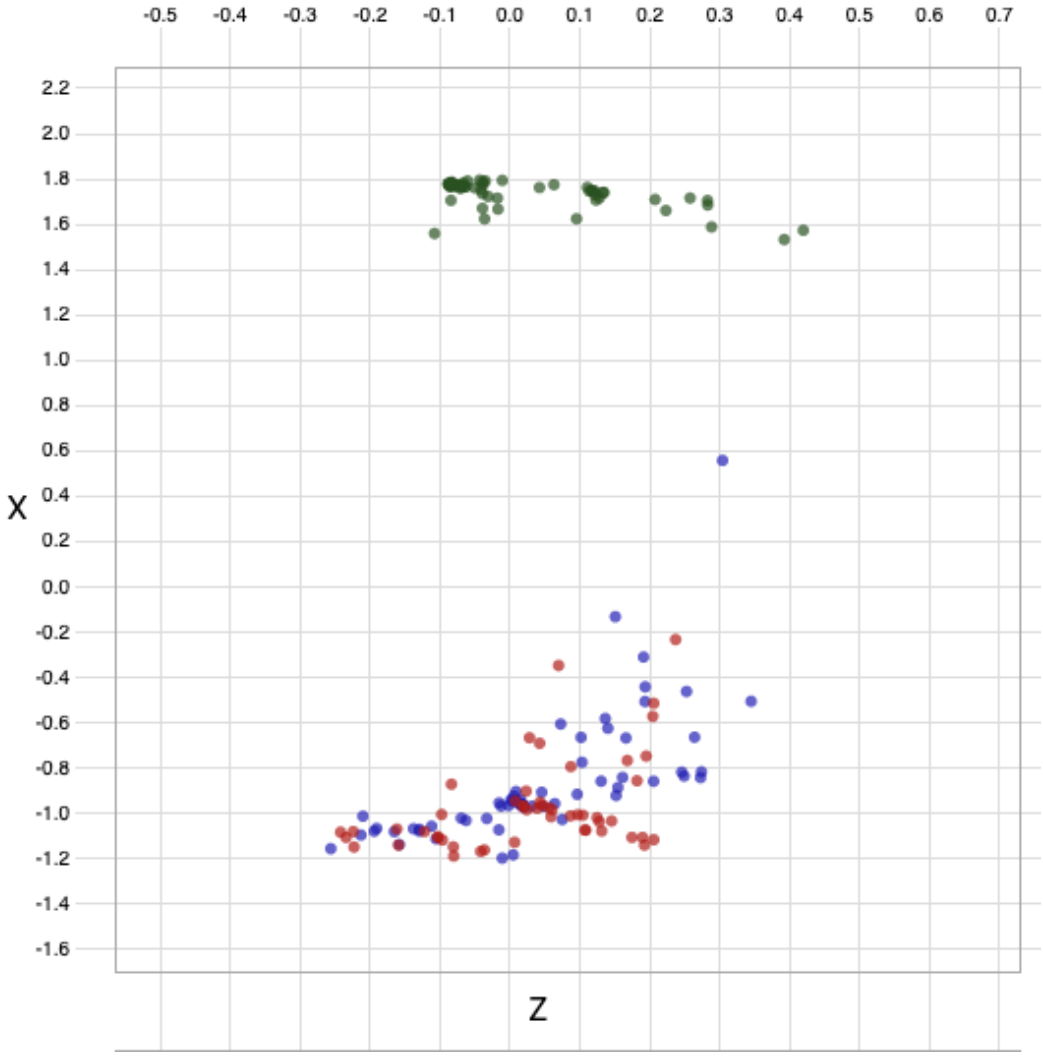



Figure 3.6: *X-Z scatter plot matrix showing the output of the classifier.The color of benign tumor is blue, malignant tumor is red and normal tumor is green.*

We are also able to show the data as specific numerical values and to compare their relationship. In our case, we can choose between the cluster values and birads and to analyze their layout and values (This is shown in figure 3.7).In the next picture, we are presented with the number of birads for each patient, and their connection with the tumor type is presented through colors. In the case that some data overlap, only one piece of information will be shown, which is the disadvantage of these plots. The 2D scatter plot provides us with the ability to showcase the density plot, with which we can see where the strongest and where the weakest intensity of the data is by using it (This is shown in figure 3.8). This display also makes brushing possible, as well as linking between different 2D scatter plots along with linking and brushing with 3D scatter

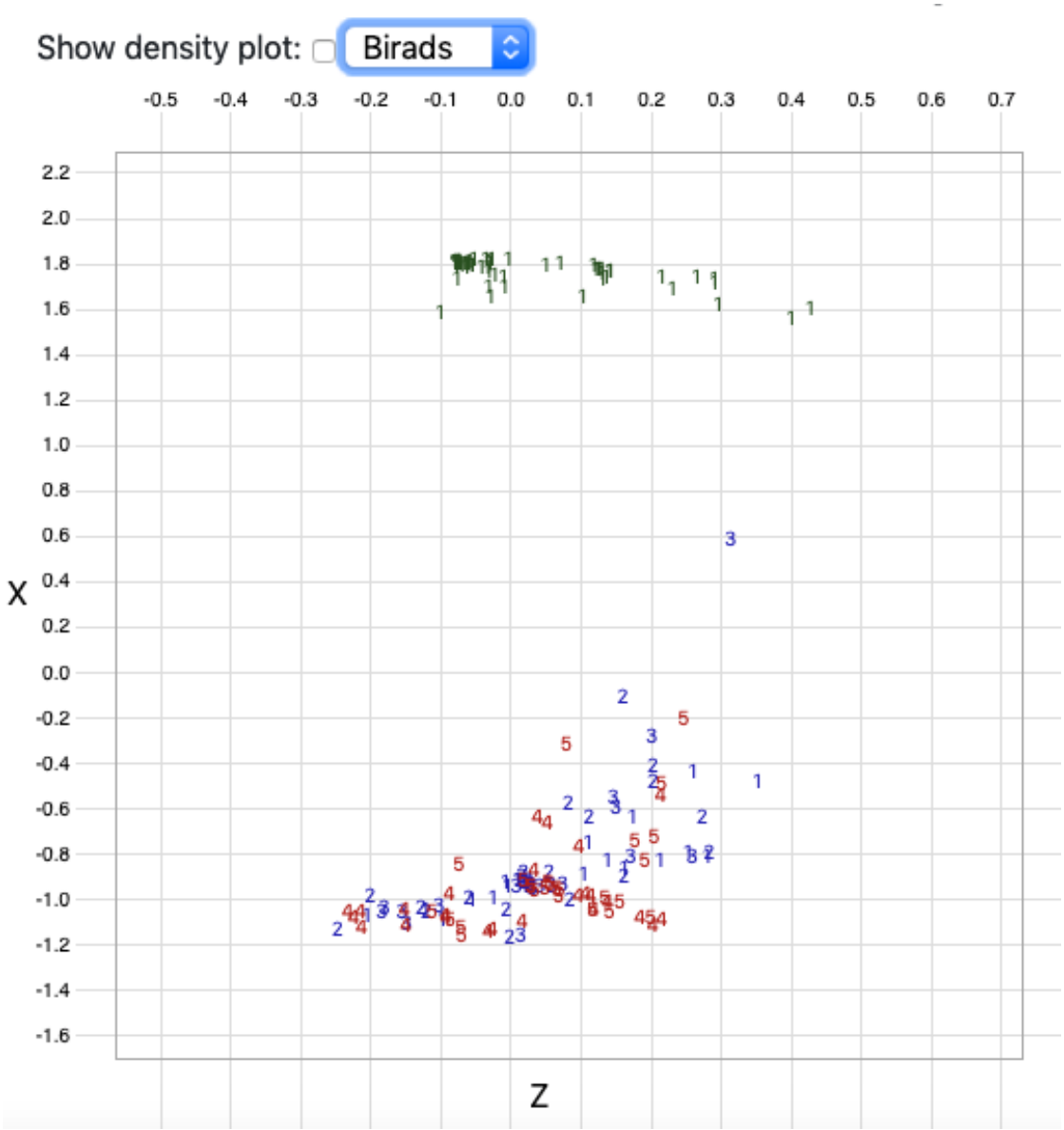



Figure 3.7: *X-Z scatter plot matrix showing the output of the classifier with birads data instead of data points.The color of benign tumor is blue, malignant tumor is red and normal tumor is green.*

plot, so we can see the relationship with other values in an easy way. The disadvantage of this display is not being able to show all three values at the same time, which is why we use the 3D scatter plot which enables us to see the connection between tumor types in 3D. By using a 3D scatter plot, we can see and compare the 3D outputs of the classifiers in one view. A big advantage as opposed to 2D is that we can compare all

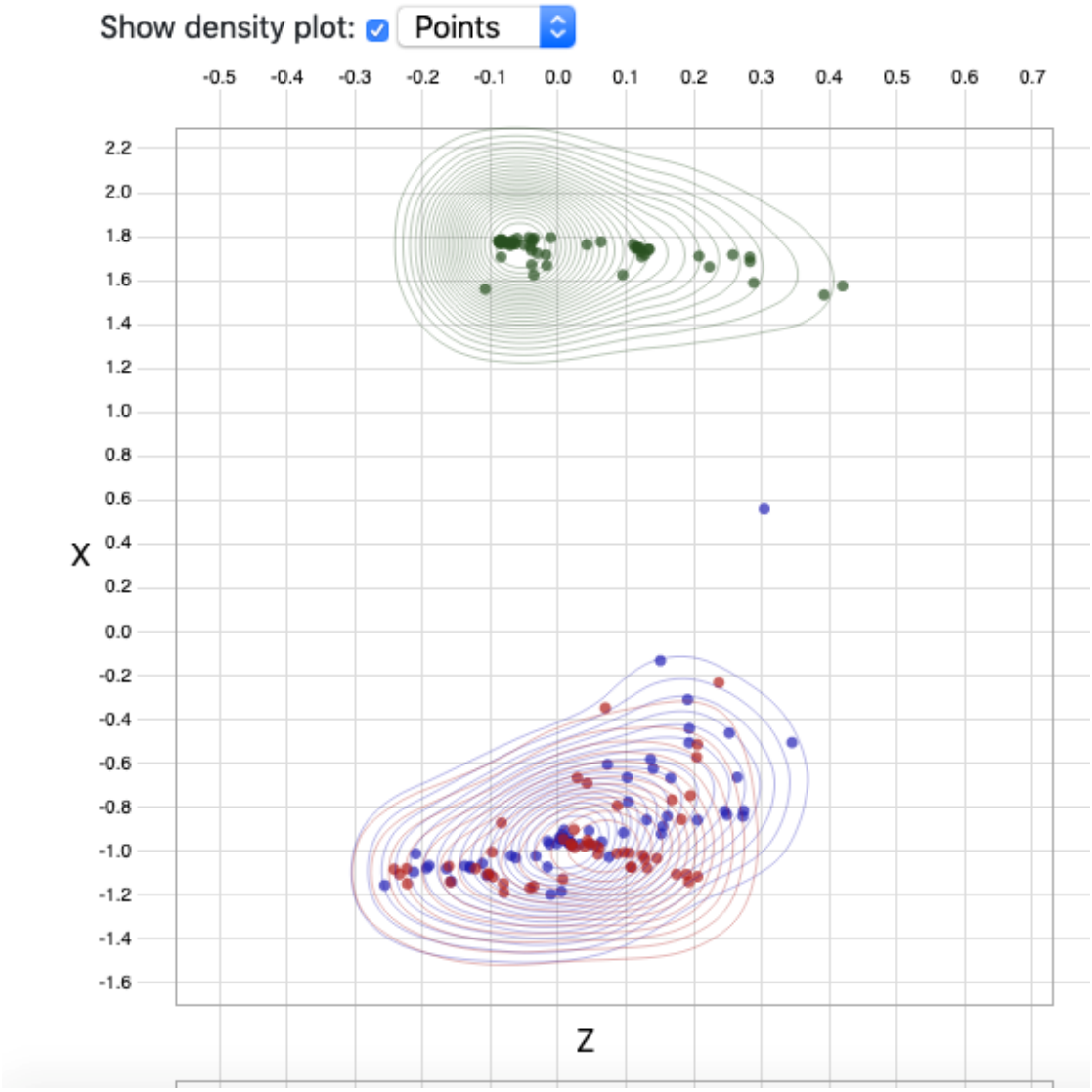



Figure 3.8: *X-Z scatter plot matrix showing density plot together with output of classifier data.The color of benign tumor is blue, malignant tumor is red and normal tumor is green.*

three values simultaneously which gives us a detailed insight into how certain groups of data are laid out. We are also presented with the possibility to look into the data from different viewpoints which is very useful. Considering the fact that different tumor types are highlighted with different colors, we are able to see which results do not match the expected results and were overlapping or deviation from expected values happened (This is shown in figure 3.9). Through our investigation, we came to the conclusion that it is very difficult to implement brushing in a 3D scatter plot because data selection must

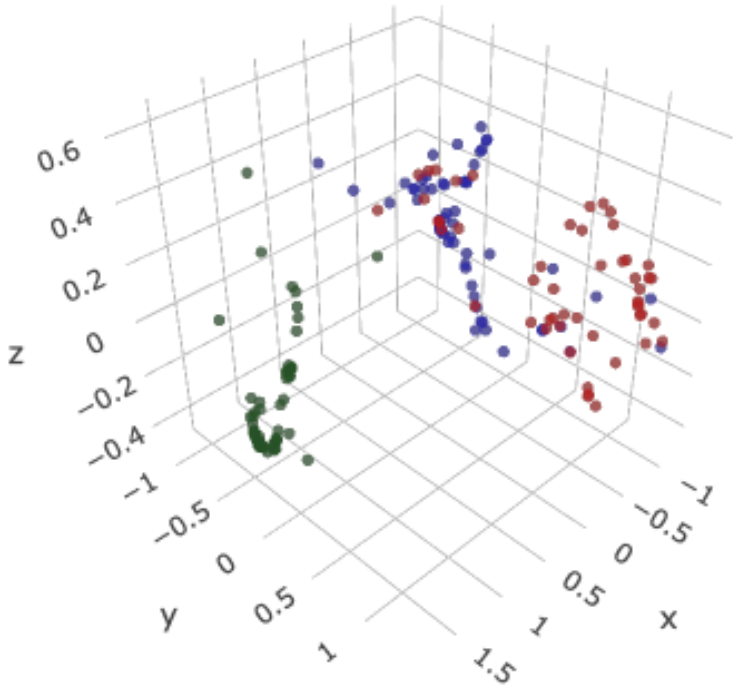



Figure 3.9: *3D representation of the output of the classifier data.The color of benign tumor is blue, malignant tumor is red and normal tumor is green.*

be multi-dimensional. Also it is very difficult to do a 3D selection on a 2D screen. This representation is also limited to a maximum of three values because each output of the classifier represents one dimension and in the case that the output of the classifier has more than three values, such a representation would not be possible.

## 3.4 Multi-class Exploration Visualization

For the multi-class exploration task we need to compare the classifier outcome with other (e.g. clinical) classification schemes. In order to represent it in the most simple way possible, we used a multidimensional Sankey plot [19], with which we can show how certain types of tumors are arranged across other groups in the chart (This is shown

in figure 3.10). Aside from this method, these data could have been presented through

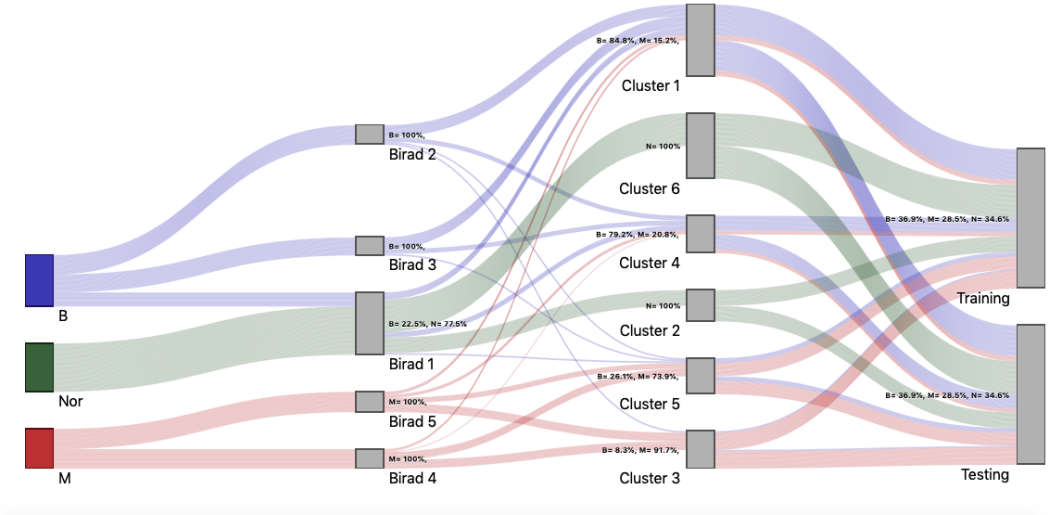
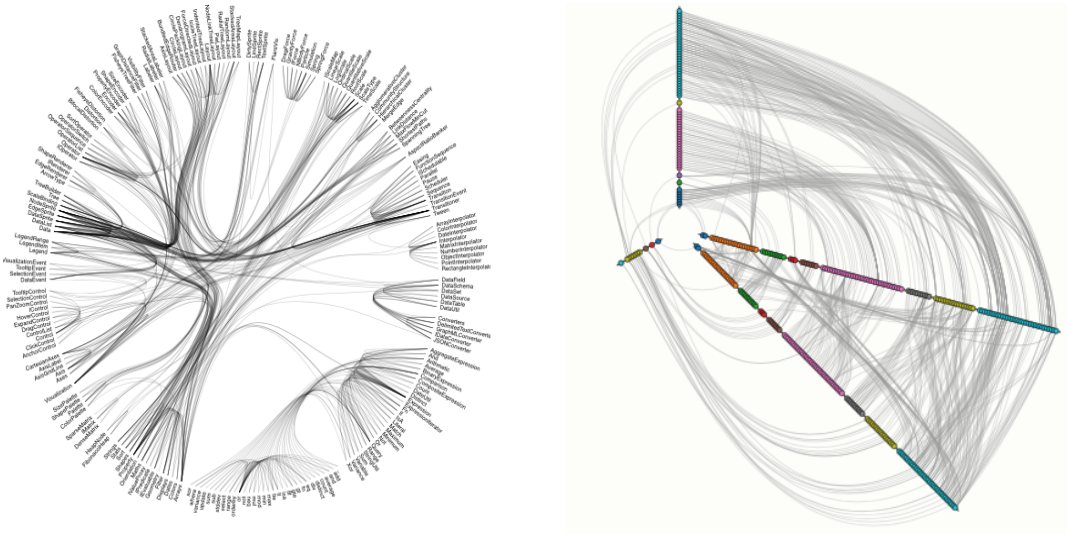



Figure 3.10: *Multi-class exploration data showing how tumor types are connected with birad, cluster and training or testing data.*

Hierarchical Edge Bundling [4, 17] or Hive plots [7, 18] (This is shown in figure 3.11) but in our case, the relationship between given values can best be seen by Sankey plot. The Sankey plot is most often used when we do not have a large number of dimensions that we need to compare, otherwise, it would be confusing to track the comparisons, across all groups . In the present case, we have a maximum of four classification schemes (tumor type, birad, cluster, training/testing) that should be compared, which is why we chose this plot.

The advantage of a Sankey plot is that we can hide or display certain dimensions depending on whether we need them. An example is shown in figure 3.12. With the Sankey plot, we can easily add new dimensions that we want to display and compare with the rest. The disadvantage is that dimensions are not horizontally reorderable, but only vertically. Next to each node in the graph, it is possible to show the percentage of the previous one of all previous nodes that had gone into it. That way we can easily compare the values between all the nodes in the graph. Each tumor type is highlighted with a color in the Sankey plot also, and it extends through the entire length of the graph so that we know at any given time and any given node what type of tumor it is. In the case of linking, the Sankey plot allows linking to other charts that we have in our web application, but linking from other plots to the Sankey is not possible due to the type and manner of displaying the data we have in the Sankey plot.



(a) *Hierarchical Edge Bundling [4, 17]* (b) *Hive plot [7, 18]*

Figure 3.11: *Sankey plot alternatives.*

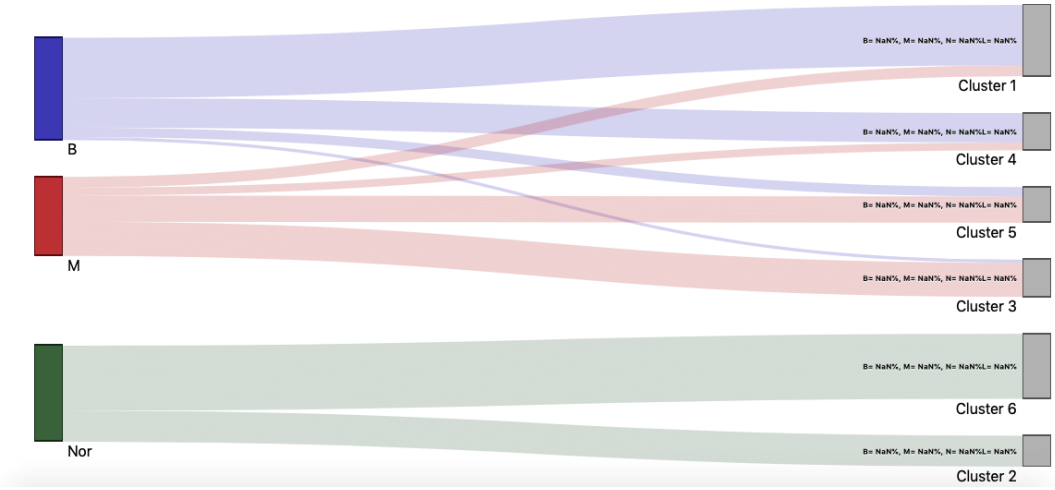



Figure 3.12: *Example of hiding unnecessary dimensions from Sankey plot.*

CHAPTER 4

# Implementation

## 4.1 Employed Technologies

The Web Application is written in Javascript with the help of the D3.v5. library. Besides, we used HTML and CSS to structure and enhance the look oft the application. We also used Frontend-CSS-Framework Bootstrap. The development environment was Atom with some plug-ins for javascript.

## 4.2 Choosing D3 version

When we started implementation there were a lot of problems when using the D3 library. The problems were mainly due to the different versions that D3 offers. We started by using the D3.v3 version, but further implementation showed that some important things we needed were not supported in the D3.v3 version (for example drawing density plot in scatter plot). After that we saw that version 4 was not very optimal either, so we decided to use D3.v5. This is also now the last version of D3 which is D3.v5.

## 4.3 Core Application

### 4.3.1 Visualization of input data of classifier

For the implementation of input data, we used the already given library **parcoords** from d3.v5. It had previously defined functions through which we need to enter the desired parameters. Firstly we want to define in the ***data()*** function which data we want to show and then we may call all other functions. By calling the ***brushMode("1D-axes")*** function we assign the ability to brush on every axes to parallel coordinates. By selecting a part of the data on a particular axis, it highlights that data and hides the other data so that you can see the selected part as much as possible. In addition to this standard

brushing function, we have defined a ***on("brushed" function(items))*** in which we define what should happen when we finish the brushing process We did this because we need the data selected in parallel coordinates to be displayed on other graphs, for example, to implement linking with other graphs. To enable parallel coordinates to be reorderable, we also called the ***reorderable()*** function in which this property was already implemented. Another important function is ***hideAxis()***, in which we define what data from our CSV file we do not want to display in parallel coordinates. The function ***alpha()*** provides us with the ability to set the opacity for each oft he vertical lines in the parallel coordinates. To include reorderabiliy of axes in parallel coordinates we also called the ***reorderable()*** function in which this property is already implemented. In addition to the already defined functions provided by this library, we had to implement a ***color()*** function that returns the color of the line for each patient, depending on what type of tumor he has. Also, with the ***bundleDimension()*** function, we enabled the grouping of similar data based on the selected parameter into the bundleDimension for easy analysis. We selected *tum_type* as the default value in bundleDimension.

### 4.3.2 Visualization of output data of classifier

The output data is presented in two ways, as 2D, and as 3D scatterplots. To display the 2D scatter we created a simple SVG, into which we then added the green x and y axes to create 3 different scatter plots. Then, through the **(circle)** function, we drew circles representing the output data for each patient. In the 2D scatter plot, we can display the density plot and for this implementation, we needed additional function ***d3.contourDensity()*** and ***d3.geoPath()***. We also used the ***d3.brush()*** function here to enable linking and brushing with other graphs. Here we define three functions ***on("start", brushstart***, ***on("brush", brushmove)***, ***on("end", brushend)*** which determines how the brush should work. The brushmove function allows after selecting one piece of data in 2D, the scatterplot selected window can move and select other values, and automatically updates the selection on other graphs For the 3D scatter plot, we used the *Plotly* Javascript Open Source Graphing Library [10, 24], as it offers us more features to manipulate data and is easier to implement. Each type of tumor presented us with a trace in plotly. We need to define a layout to put all the created traces in one array. Then, by calling the function ***Plotly.newPlot (graphDiv, data_scatter, layout)***, we create a 3D data view. *GraphDiv* represents the HTML element into which a 3D plot should be created. We used a politely click event to allow the selection of individual points in a 3D plot for further linking with other graphs. Plotly, unfortunately, does not yet have an implemented function that allows us to select multiple points at once.

### 4.3.3 Multi-class exploration visualization

We used a Sankey plot to display this information. The Sankey plot consists of nodes and links. Here we also used simple SVG to which we further added elements. Nodes are represented through rectangles created by the append function *rect* and links are lines that connect specific nodes. The ***on("drag")*** function allows us to vertically position

and move the nodes. Each node displays the percentage of nodes that enter it that is implemented in the ***showPrecentage()*** function and is displayed on the node with ***append("text")*** function.

### 4.3.4 Interaction

The interaction between the plots is done by brushing and linking [1, 2]. In D3 we already have implemented brushing methods and we use them for selecting some data in our representations. We can select multiple data at once with the help of linking show this data on other plots. Linking is implemented with CSS so that we show or hide data that are selected. In the case of the 3D plot, where we use Plotly, we have implemented brushing ourselves. As I mentioned before, brushing works by selecting point into a 3D scatter plot one by one. Linking is also in this case implemented with CSS.

CHAPTER 5

# Results

## 5.1 Exploration of Input and Output of Classifier

By visualizing the input and output of the classifier, we made it possible to compare one or more types of tumors based on selected criteria. In most cases, we are interested in the types of tumors that have different values than all other values of that tumor. Through a few examples, we will show how such points can be easily found and compared through our visualization. The following figure shows the input of the classifier. All patients who have a normal tumor are shown in green lines and have similar values for all parameters except for two patients who have different values for *calc_variance* values compared to other patients. Therefore, we will select these two patients and analyze them through their output values(This is shown in figures 5.1 and 5.2).

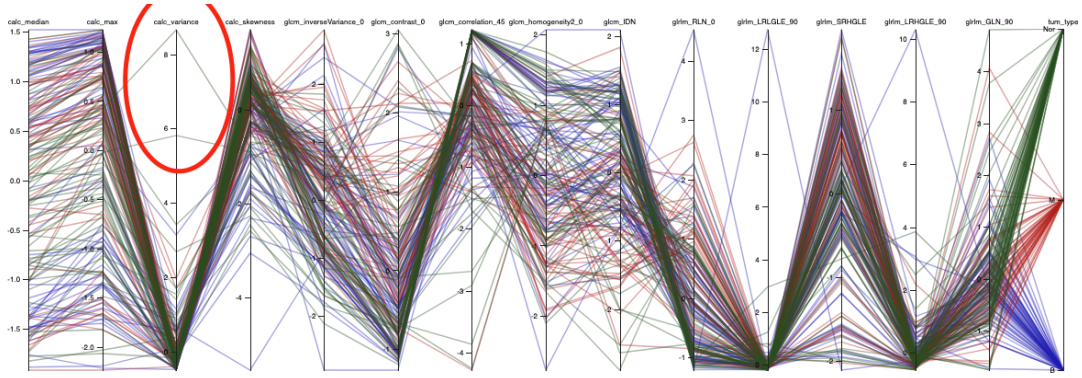



Figure 5.1: *Parallel coordinates with no data selected. In the red circle are patients that we chose for selecting. In figure 5.2 we can see the same image with these two patients selected.*

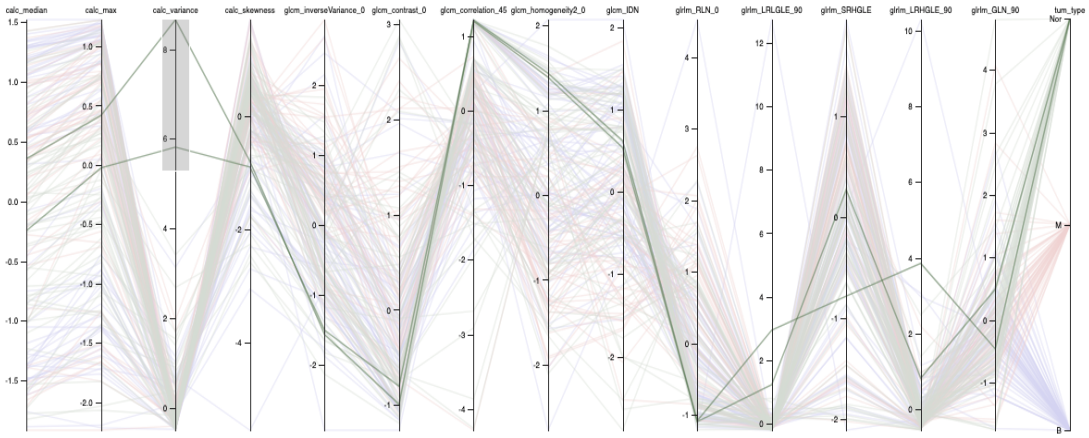



Figure 5.2: *Parallel coordinates showing two selected patients.*

Using the bundling option offered by our implementation, we can group together data that have similar values and thus get a clearer picture which the points of interest for analyzing may be. This option may have performance issues if we have data on a large number of patients. In our case, we chose to group the bundling into that of the tumor type and get the following results (This is shown in figure 5.3).

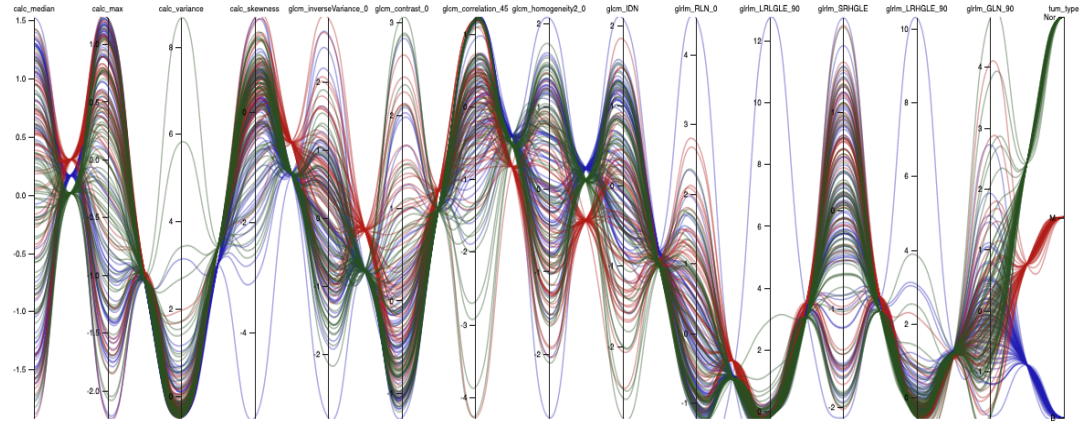



Figure 5.3: *Applying bundling to parallel coordinates.*

This figure shows even more clearly what values deviate from some predicted ones in our case on the *calc_varinace* axis. We see two patients with normal tumors who do not belong to the bundling group for normal tumor values. By selecting these two patients, we obtain their output values on a 2D and 3D Scatter plot that can be further analyzed (This is shown in figures 5.4 and 5.5). The figure on the left shows the output data before selecting two selected patients, and the image on the right shows two patients selected in

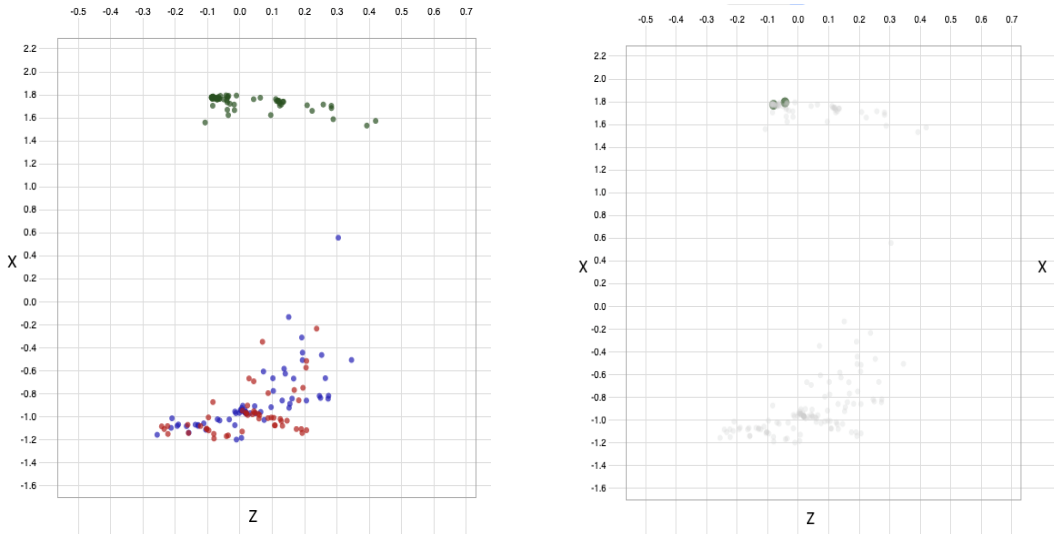



Figure 5.4: *Showing how selecting data in 2D scatter plot works.*

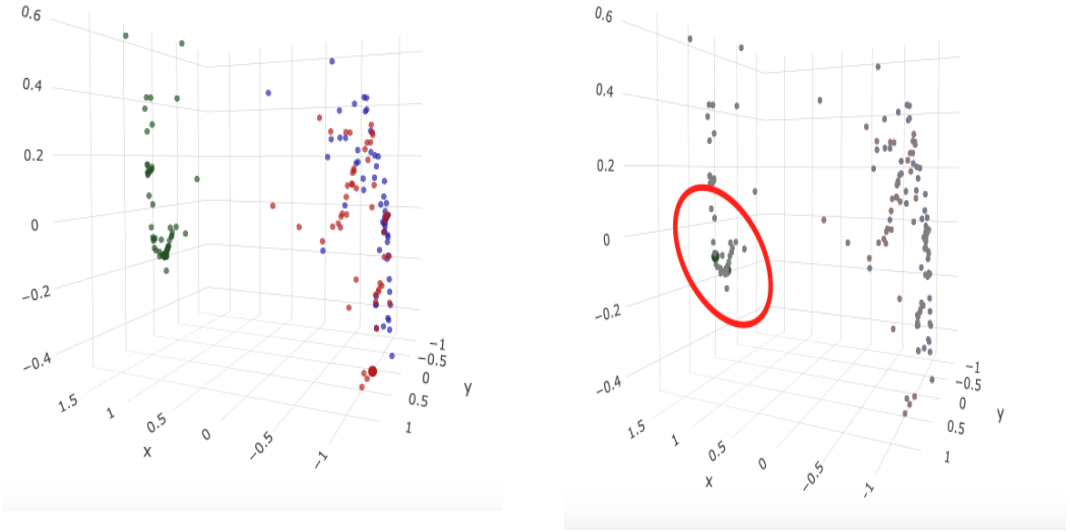



Figure 5.5: *Showing how selecting data in 3D scatter plot works.*

parallel coordinates. As seen in the figure in the case of the output of the classifier, these two patients are in the group with all other patients who have a normal tumor. This way we have shown that the *calc_variance* value from the input of classifier does not affect the results in the output of the classifier. We also see through the 3D view that these two patients are in a group with other patients with normal tumors. As we have compared the input values with the output values, we can do the opposite. By selecting the points of interest to us on a 2D or 3D scatter plot, we can display them in parallel

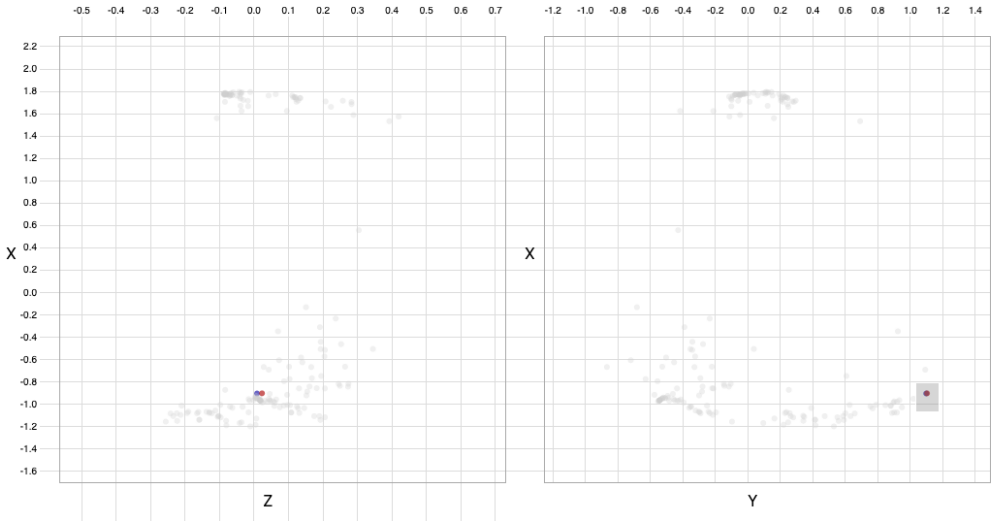



Figure 5.6: *Selecting two patients with almost identical output data in 2D.*

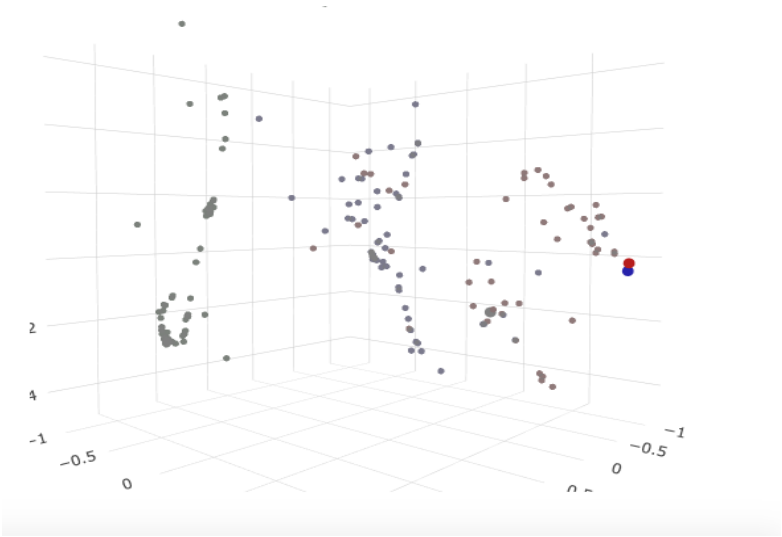


Figure 5.7: *Selecting two patients with almost identical output data in 3D.*

coordinates and analyze their behavior. The following two patients have almost identical outputs (This is shown in figures 5.6 and 5.7), although one has a benign and the other a malignant tumor. By selecting these two patients, we analyze how much their values differ at the input and we conclude that despite the identical values for the output, there are two different tumors. In the 3D view, we can see even more clearly how similar the parameters for the two patients are. Possible problems that may arise when creating a 3D scatter plot are a large amount of data that needs to be entered into the 3D image.

Analyzing the display in parallel coordinates, we can see that the values on all axes

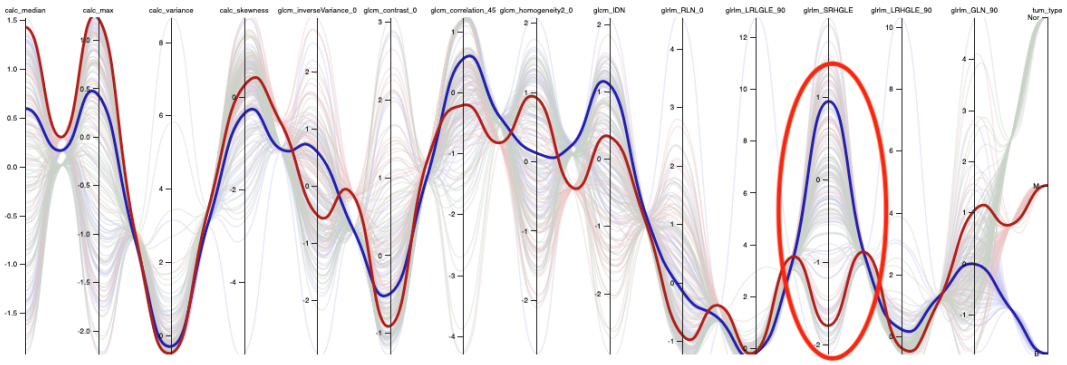



Figure 5.8: *Exploration input and output data. We see in picture differences between two selected patient.*

are almost identical except on the *grlrm_SRHGLE* axis where the two values differ drastically. Based on these data, we can conclude that the parameter *grlrm_SRHGLE* is important for the factor that determines which type of tumor is involved (This is shown in figure 5.8).

We can also analyze output values based on birad or cluster type. Thus, we can conclude that some of these values affect the value input. The following figure shows the layout of the output data based on the cluster type. From the picture, we can see that the same clusters are grouped together, which in the case of birads does not apply, as we see in the following picture (This is shown in figures 5.9 and 5.10). From the first picture showing

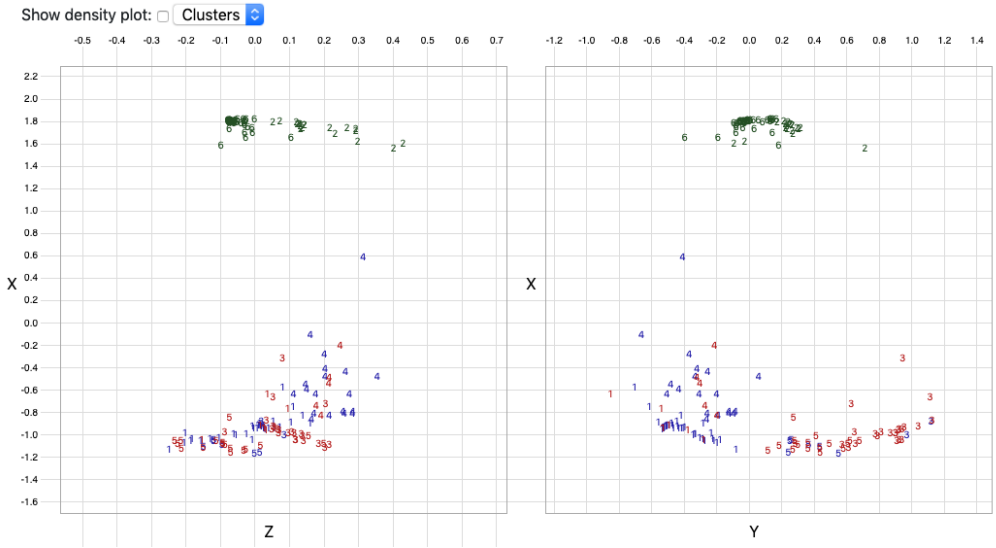



Figure 5.9: *Analysing data based on birads.*


Figure 5.10: *Analysing data based on cluster type.*

the values for the cluster, we selected two values that have the same cluster and about the same values for *x, y, z* to analyze their input values. Here we see that a big difference appears with the values for the *calc_median* axis, which leads us to the conclusion that this value can be closely related to the cluster type (This is shown in figure 5.11).

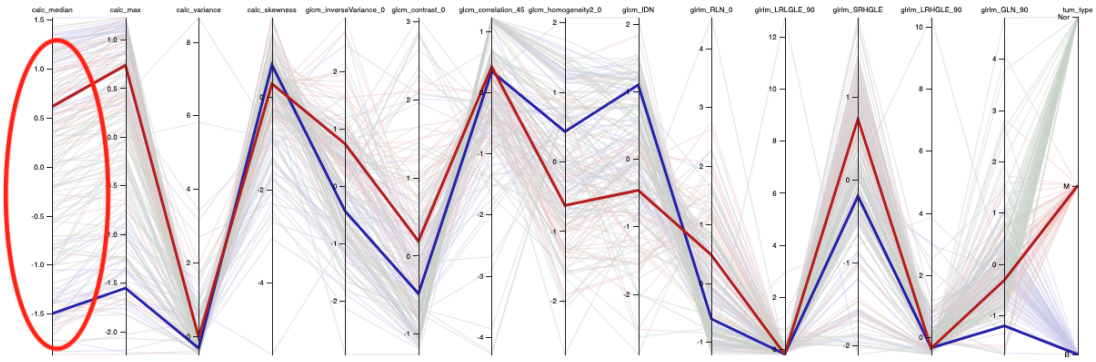


Figure 5.11: *Input data analysing base od selected data from output.*

## 5.2 Multi-class Exploration and Analysis

With Multi-class Exploration, we can show how much a part of the tumor affects a particular birad or cluster and how much patient data were training data and how much testing data (This is shown in figure 5.12). In the following example, we can see that the Birad with value 1 consists of 22.5% benign tumor and 77.5% normal. We can see the

same for every cluster value. E.g. Cluster 1 consists of 88.8% benign tumor and 12.2% malignant tumors. This means that 88% of patients who have a value for 1 have a benign tumor. Another interesting information we see from the picture is that all patients who have a cluster value of 6 or 2 have a normal tumor and if they have a value of cluster 3 they have a malignant tumor.

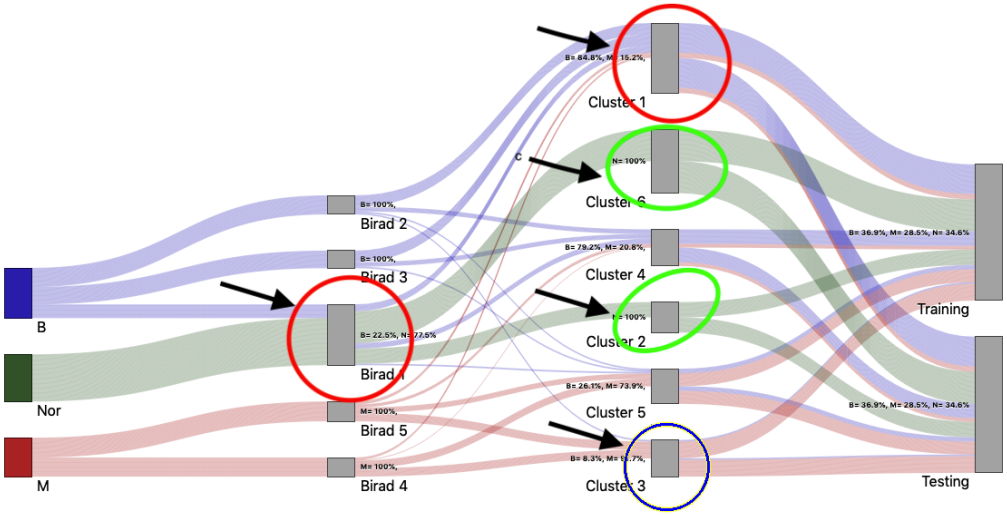



Figure 5.12: *Sankey plot with all classification schemes.*

If we want to omit one of the variables in order to have a clearer picture of how the tumors are arranged, by clicking on the checkbox we can choose what values we want to see in the Sankey plot (This is shown in figure 5.13). In the following example, we omitted birad and cluster data to see more clearly how much tumor data was tested and how much was training data.

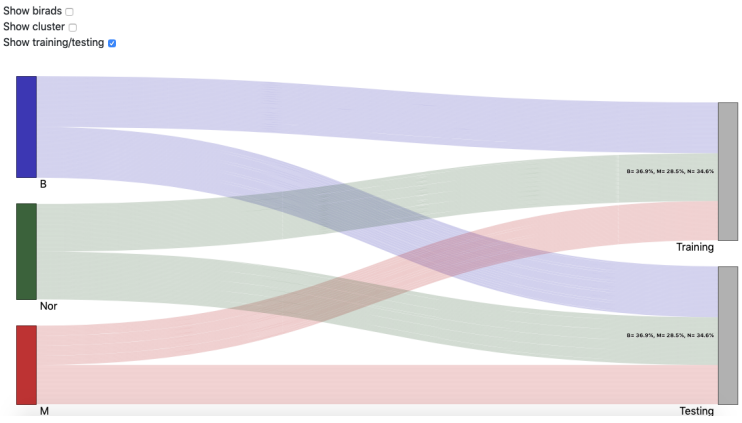



Figure 5.13: *Showing results without birad and cluster dimensions.*

It can be clearly seen from the picture that each tumor has half the training data and half the test data. If we want to show only the patients who have done the testing on the other plots, by clicking on the node *Testing* we automatically show that data on the other plots.

CHAPTER 6

# Conclusion

## 6.1 Summary

Breast cancer classification can be supported by multivariate data visualizations. With our representation, researchers developing novel tumor tissue classifier can better understand and explore their data. Using D3 and Plotly we have developed a strategy to support classification exploration and analysis. In our web application, the user can view four different graphs of tumor data classification. Based on the information obtained, the user can generate a new hypothesis for some further research in the future.

In Chapter 3 (Methodology) we discussed why we used individual visualizations for a particular type of data. We presented the advantages and disadvantages of each visualization and talked about their limits and alternatives. We have shown why our representations are better than others for the presentation and analysis of breast cancer data.

In Chapter 4 (Implementation) we discussed libraries and code dependencies. We explained what technologies we used to implement this web application and why we used them. Besides that, we analyzed and explained some interest cases in our implementation.

Results and analysis of visualizations are presented in Chapter 5 (Results). We tested our application from the perspective of a future user and displayed the results through a usage scenario. Through the results, we demonstrated all the functionality of our application. We also discussed possible performance problems.

All in all, our application is a promising basis that offers new exploratory opportunities for researchers working on tumor tissue classification, and in many cases, this can help researchers to speed up and better analyze their data.

## 6.2 Future Work

In future work, we can concentrate on adding more data about patients and representing their data through additional visualizations. For example, if we add additional parameters like Birad or cluster, we should employ additional representations for these data. We can also create new extensions where we can add medical imaging data. From these images, we can analyze the underlying medical imaging data and make graphical representation for analyzing interesting pathological information with the patient image. It will be good to create some representations where the user can have an anatomical view on data of a specific patient. Moreover, there is still space for improvement of the 3D view. Linking from a 3D view can also be improved.

On the other side, we can focus our future work on the creation of a web application where we can compare visualizations from multiple data sets at the same time. With that, we can observe some changes, similarities, and differences of data from different countries, time and so on. In addition, a further implementation could be based on an extension that would allow users to perform a comparison of many different classifiers.

# Bibliography


[1] R. A. Becker and W. S. Cleveland. Brushing scatterplots. *Technometrics*, 29(2):127–142, 1987.

[2] A. Buja, J. A. McDonald, J. Michalak, and W. Stuetzle. Interactive data visualization using focusing and linking. 91:156–163, 1991.

[3] D. B. Carr, R. J. Littlefield, W. Nicholson, and J. Littlefield. Scatterplot matrix techniques for large n. *Journal of the American Statistical Association*, 82(398):424–436, 1987.

[4] T. Dang and A. Forbes. Cactustree: A tree drawing approach for hierarchical edge bundling. In *2017 IEEE Pacific Visualization Symposium (PacificVis)*, pages 210–214. IEEE, 2017.

[5] H. Doleisch. Simvis: Interactive visual analysis of large and time-dependent 3d simulation data. In *2007 Winter Simulation Conference*, pages 712–720. IEEE, 2007.

[6] H. Doleisch, M. Gasser, and H. Hauser. Interactive feature specification for focus+ context visualization of complex simulation data. In *VisSym*, volume 3, pages 239–248, 2003.

[7] S. Engle and S. Whalen. Visualizing distributed memory computations with hive plots. In *Proceedings of the Ninth International Symposium on Visualization for Cyber Security*, pages 56–63, 2012.

[8] D. L. Gresh, B. E. Rogowitz, R. L. Winslow, D. F. Scollan, and C. K. Yung. Weave: A system for visually linking 3-d and statistical visualizations applied to cardiac simulation and measurement data. pages 489–492, 2000.

[9] D. Holten and J. J. Van Wijk. Force-directed edge bundling for graph visualization. In *Computer graphics forum*, volume 28, pages 983–990. Wiley Online Library, 2009.

[10] P. T. Inc. Collaborative data science : https://plot.ly, last accessed: 15/02/2020.

[11] A. Inselberg. The plane with parallel coordinates. *The visual computer*, 1(2):69–91, 1985.

[12] A. Inselberg and B. Dimsdale. Parallel coordinates for visualizing multi-dimensional geometry. In *Computer Graphics 1987*, pages 25–44. Springer, 1987.

[13] C. Kai. Brushable scatterplot matrix : https://observablehq.com/@d3/scatterplot-matrix, last accessed: 15/01/2020.

[14] C. Kai. Parallel coordinates : https://syntagmatic.github.io/parallel-coordinates/, last accessed: 21/02/2020.

[15] A. Kamra, V. Jain, S. Singh, and S. Mittal. Characterization of architectural distortion in mammograms based on texture analysis using support vector machine classifier with clinical evaluation. *Journal of digital imaging*, 29(1):104–114, 2016.

[16] K. Kerlikowske, D. Grady, J. Barclay, V. Ernster, S. D. Frankel, S. H. Ominsky, and E. A. Sickles. Variability and accuracy in mammographic interpretation using the american college of radiology breast imaging reporting and data system. *Journal of the National Cancer Institute*, 90(23):1801–1809, 1998.

[17] B. Mike. Hierarchical edge bundling : https://observablehq.com/@d3/hierarchical-edge-bundling, last accessed: 20/02/2020.

[18] B. Mike. Hive plots, 2012 : https://bost.ocks.org/mike/hive/, last accessed: 20/02/2020.

[19] B. Mike. Sankey diagram : https://observablehq.com/@d3/sankey-diagram, last accessed: 20/02/2020.

[20] P. Muigg, J. Kehrer, S. Oeltze, H. Piringer, H. Doleisch, B. Preim, and H. Hauser. A four-level focus+ context approach to interactive visual analysis of temporal features in large scientific data. 27(3):775–782, 2008.

[21] S. Obenauer, K. Hermann, and E. Grabbe. Applications and literature review of the bi-rads classification. *European radiology*, 15(5):1027–1036, 2005.

[22] S. Oeltze, H. Doleisch, H. Hauser, P. Muigg, and B. Preim. Interactive visual analysis of perfusion data. *IEEE transactions on visualization and computer graphics*, 13(6):1392–1399, 2007.

[23] S. G. Orel, N. Kay, C. Reynolds, and D. C. Sullivan. Bi-rads categorization as a predictor of malignancy. *Radiology*, 211(3):845–850, 1999.

[24] C. Sievert, C. Parmer, T. Hocking, S. Chamberlain, K. Ram, M. Corvellec, and P. Despouy. plotly: Create interactive web graphics via 'plotly. js'. *R package version*, 4(1):110, 2017.

[25] P. Tukey and J. Tukey. Graphic display of data sets in 3 or more dimensions. *The collected works of John Tukey*, 5:189–288, 1988.