NOVA IMS Information Management School

MAA

Master Program in Advanced Analytics

Can machine learning methods contribute as a decision support system in sequential oligometastatic radio-ablation therapy?

Marius Wilhelm Löwe

Thesis presented as the partial requirement for obtaining a Master's degree in Data Science and Advanced Analytics

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação
Universidade Nova de Lisboa

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação
Universidade Nova de Lisboa

Can machine learning methods contribute as a decision support system in sequential oligometastatic radio-ablation therapy?

by
Marius Löwe

Thesis presented as the partial requirement for obtaining a Master's degree in Data Science and Advanced Analytics

Advisor: Leonardo Vanneschi

May 2021

Abstract

Cancer treatment is among the major medical challenges of this century. Sequential oligometastatic radio-ablation (SOMA) is a novel treatment method that aims at ablating reoccurring metastasis in a single session with a targeted high dose of radiation. To know if SOMA is the best possible treatment method for a patient, the benefits of each available therapy need to be understood and evaluated.

The ability to model complex systems, such as cancer treatment, is the strength of machine learning techniques. These techniques have improved the understanding of numerous medical therapies already. In some cases, they can serve as medical support systems if they deliver reliable results that doctors can trust and understand.

The results obtained from applying numerous machine learning techniques to the data of SOMA-treated patients show that there are favorable techniques in some cases. It was observed that the Random Forest algorithm proved superior at different classification tasks. Additionally, regression problems opposed a great challenge, as the amount of data is very limited. Finally, SHAP values - a novel machine learning interpretation technique – provided valuable insights into understanding the rationale of each algorithm. They proved that the machine learning algorithms could learn patterns aligned with the human intuition in the problems presented.

SHAP values show great potential in bridging the gap between complex machine learning algorithms and their interpretability. They display how an algorithm learns from the data and derives results. This opens up exciting possibilities for applying machine learning algorithms in the real world.

Table of Contents

Table of figures

Table of tables

1. Introduction

In this chapter, after a brief introduction to the underlying topic, the problem statement and the research goal of this thesis are defined. This thesis was conducted as part of a joint project with the radiation oncology department of the Champaulimaud Foundation in Lisbon, Portugal. The data is derived from patients with oligometastatic cancer treated with a specific high-precision image-guided radiotherapy.

Cancer treatment is among the most significant medical challenges of this century, as cancer is responsible for about 10 million deaths globally per year. Therefore, according to the WHO, it is the second leading cause of death. Radiotherapy is, next to chemotherapy and surgery, among the few treatments available. Selecting the best treatment to cure the patients is the doctor's task. However, while the cure is always the desired goal, other factors such as the patient's quality of life or prolonging the patient's life have to be considered by the doctors as well. With an increased amount of data collected from cancer patients in recent years, machine learning can potentially offer additional value to the decision process at various stages of the therapy.

Radiomics is a rapidly emerging technique in radiology that has enabled new radiotherapy methods by extracting and modelling three-dimensional data from radiological images with artificial intelligence (Kumar et al., 2012). Furthermore, artificial intelligence is used to develop predictive or descriptive models from the data obtained. In Radiomics, it is believed that extracting information from medical images, often Positron-Emission Transmission Computer-Tomography (PET-CT) images, provides additional diagnostic or predictive information that escapes the human eye and will complement the information available to the radiologist (Cook et al., 2014). Only the recent developments in technology, biomarkers, and computer-assisted-detection systems (Jansen et al.) have enabled this method (Philippe Lambin et al., 2012). These recent developments have not only improved diagnoses and treatment but led to novel therapeutic approaches.

Sequential oligometastatic radio-ablation (SOMA) therapy is a novel cancer treatment method applied by the radiation oncology department of the Champaulimaud Foundation, which is based on the same technology that enabled radiomics. It utilizes high doses of very targeted radiation in a single ablation session (SDRT) to kill sequentially arising cancer metastases. This therapy method requires a highly technical setup to successfully deliver the high doses of radiation to the desired destination. Whether machine learning methods can help improve the understanding and provide decision support for SOMA therapy is the rationale of this thesis.

1.1 Problem statement

New medical treatment methods must be well researched, evaluated, and understood before addressing a broad base of patients. Mistakes are usually penalized with reduced patients' wellbeing. Implementing machine learning methods in a medical context can lead to better understanding and learn complex connections. However, especially in the context of a novel technique, such as SOMA, it brings different levels of complexity with it:

Firstly, the data availability is low. With novel treatment methods, studies on just a few patients have to prove the benefits of the treatment method over the well-established methods. Furthermore, the technical facilities required for this treatment method are very advanced, restricting access to a small patient group. Additionally, in the field of radiomics, where measurements of the cancer cell are taken, the variability between different machines reduces the potential availability of consistent data.

Secondly, the quality of the data is limited. The human organism is very complex and has been attempted to be fully understood for many centuries. Numerous factors affect cancer treatment success, such as the genome, the patients' medical history, and even nutrition. In a perfect machine learning setting, all influencing factors would be included in the data. However, this is not feasible, as some of these factors cannot be explained yet, are not available, or simply, doctors do not have the time to collect them. Additionally, the imbalanced nature of the data poses a problem for machine learning. Treatment is advancing, producing increasingly better results. This opposes the challenge that there is a decreasing amount of unsuccessful treatments. This is great from a medical standpoint, however, when classifying successful and unsuccessful treatments, this results in difficulties training machine learning models.

Thirdly, data of patients' treatment history is often incomplete. In the case of long-lasting cancer treatment therapies, patient data is collected over several years. However, patients can change doctors, not attend follow-up examinations, move away or die without notice to the treating doctors. This is challenging as it might lead to noise, missing or incorrect data. Unfortunately this is impossible to detect in the data and requires better generalization ability of the model.

Finally, doctors need to trust and derive insights from the machine learning methods to include this in their work with their patients. Most machine learning methods are novel, especially for medically trained people. The probability is high that they have never heard of them. The challenge is that the doctor in charge of treatment needs to trust the insights generated by machine learning enough to treat the patient based on this knowledge. This is especially difficult with "black box" algorithms that offer no insights on how they arrived at a particular conclusion.

1.2 Research goal

The doctors who treated the patients with SOMA therapy as part of their cancer treatment process derived several variables for further analysis. The data was collected over the timespan of patients' treatments. They describe the patient's treatment process over a specific period. Generating insights into a specific treatment's immediate success or a better understanding of the patients' treatment path is desired.

The research goal of this thesis is to identify whether machine learning methods can create value within the boundaries of the problems and the data outlined previously. At various points in time during a patient's treatment process, decisions have to be made with varying degrees of uncertainty. Supporting these decisions with analytically advanced methods potentially directly impacts the patients' quality of life. In this context, the motivation for applying machine learning algorithms is to potentially model complex relationships that escape univariate analysis. Providing a starting point on which machine learning methods perform well to solve the stated problems and whether they can create value in the treatment process is the goal of this thesis. A particular focus is set on various machine learning algorithms' performance and their interpretability through a novel concept of model interpretability.

To achieve the research goal, six dependent variables are forecasted with the patients' treatment data. Two variables are associated with the cancer lesion and four specific to the individual patient. Algorithms will be compared to identify the algorithms delivering the best performance in the environment of the data. Furthermore, their interpretability potential is lifted by applying the concept of SHapley Additive exPlanations (Mokhtari, Higdon, & Başar, 2019). This relatively new machine learning technique aims at increasing model interpretability through a game-theoretical approach (Lundberg & Lee, 2017).

2. Related Work

This thesis attempts to evaluate to what degree machine learning methods can impact the decision-making process of treating patients with SOMA (sequential oligometastatic radio-ablation therapy). Various supervised machine learning techniques are compared to each other and evaluated on different problems stated in SOMA therapy. This section provides a review of the current status of research conducted in this context. The three fields discussed in this thesis are radiomics, supervised machine learning, and SHAP values.

2.1 Radiomics

Cancer is a heterogeneous disease characterized by various subtypes, degrees of invasiveness, influencing factors, and location. This heterogeneity of the disease is equalled in diagnosis parameters and treatment methods. With increasingly more technical advances in cancer treatment methods in recent years, the data availability has increased (El Houby, 2018). This increasing amount of cancer data has raised the interest of data mining and machine learning researchers because of its high degree of complexity, data types, and variability. This resulted in the emergence of Bioinformatics, which combines the statistical and machine learning knowledge in a biomedical context, initially rooted in sequencing genes with computational power (Dayhoff & National Biomedical Research, 1969).

Radiomics is one of the technical advances in Bioinformatics that has proven once more the value of machines in medical treatment processes (P. Lambin et al.). Especially in oncology radiomics are advanced, here the underlying hypothesis is that medical radiographic images contain detailed and valuable information about the nature of a cancer lesion (J. Wu, Tha, Xing, & Li). Extracting the information to the full extent and making it available to base decisions on is the nature of radiomics. The method of radiomics is extracting numerous features from medical radiographic images and utilizing pattern recognition abilities of algorithms to extract information (P. Lambin et al.). Among other attributes of a tumor, it was proven that radiomics could utilize radiographic data better than the radiologist's eye in determining the heterogeneity of a tumor (Gillies, Kinahan, & Hricak, 2016) (Cook et al., 2014).

High single-dose radiotherapy (SDRT), where a high dose of radiation can be delivered accurately to the tumor in a single session was enabled by applying radiomics (Zelefsky et al.). Cancer lesions can be visualized graphically by extracting features of positron emission scans (PET /CT) (Grosu et al.). This allows the extraction of the necessary information for SDRT. Studies have found this superior over other therapy approaches in certain circumstances (Zelefsky et al.). Sequential oligometastatic radio ablation therapy (SOMA) is one of these circumstances, where the ablation of metastatic lesions with high radiation doses up to 24Gy (grey) is deemed beneficial (Greco et al., 2019).

Recently, machine learning methods gained relevance in the field of biomedical research. The understanding of machine learning models is improved by additional machine learning methods, aiming at fostering confidence in “black-box” models. Hence, they can be used to

support decisions in a medical context (Jansen et al.).It was proven that through the application of radiomics, associations to various parameters of a tumor, such as the aggressiveness, could be modeled or forecasted (Vallières, Freeman Cr Fau - Skamene, Skamene Sr Fau - El Naqa, & El Naqa),(Liu et al.).

2.2 Supervised Machine Learning

Machine learning is defined as "the study of computer algorithms that improve automatically through experience" by Tom Mitchell (Mitchell & McGraw-Hill, 1997). Machine learning is considered a subdomain of artificial intelligence, defined as "machines gaining intelligence from data without being explicitly programmed" (Samuel, 1959). While these two terms are specific to the intelligence of a system, the term data mining describes the process of extracting knowledge from more significant amounts of data (Lovell, 1983).

In their essence, these three domains overlap heavily. They are frequently applied conjunctively to obtain more information from data that may not be obtained by other analysis, statistical methods, or inspection of the human eye. Researchers are applying data mining and machine learning techniques in the biomedical context to gain more information about diseases, their treatment methods and make predictions to support their decision-making in the treatment process (Liao & Lee, 2002).

In this thesis, supervised machine learning techniques are applied, as the data provides labels for the variables that are to be predicted. Data that does not provide labels requires unsupervised machine learning techniques (Mohri, Rostamizadeh, & Talwalkar, 2018). There are two types of supervised machine learning problems; classification, and regression. In classification problems, an algorithm attempts to predict the affiliation of samples to two or more classes. In regression problems, algorithms predict a numerical value.

sThe "No Free Lunch Theorem" states that for all possible problems, all machine learning algorithms perform equally (Wolpert & Macready, 1996). However, this does not exclude the possibility of specific algorithms performing better than others in certain conditions. Numerous studies have been conducted comparing several machine learning algorithms in various contexts, including the medical (Vanneschi et al., 2011) (Uddin, Khan, Hossain, & Moni, 2019) (Tan & Gilbert, 2003). While some could observe specific favorable algorithms for the investigated problem (Vanneschi et al., 2011), others identified different algorithms in very similar but not identical settings (Ahmad, Eshlaghy, Poorebrahimi, Ebrahimi, & Razavi, 2013).

2.3 SHAP values

SHapley Additive exPlanations – SHAP values – were first introduced in the field of machine learning by Lundberg in 2017 (Lundberg & Lee, 2017). They are based on a game-theoretical approach of Lloyd Shapely (Shapley, 1953). Lundberg proposed this as an alternative, unified approach to better understand and interpret the results of machine learning models. This could mitigate the trust issue that "black-box" models can bring with them, preventing them from practical application.

Other approaches to interpreting specific machine models and model agnostic approaches have been presented previously (Ribeiro, Singh, & Guestrin, 2016; Shrikumar, Greenside, Shcherbina, & Kundaje, 2016). However, SHAP values have outperformed these other techniques when assessing the explanation of class differences and consistency with the human intuition (Lundberg & Lee, 2017).

As SHAP values are model agnostic, they are applied in various backgrounds to interpret model predictions. Applications can be found in financial applications, gene expression, and traffic security (Mokhtari et al., 2019) (Bi et al., 2020) (Parsa, Movahedi, Taghipour, Derrible, & Mohammadian, 2020). In essence, every predictive model would have the possibility to be interpreted applying SHAP values.

SHAP values are already applied in gene expression and sequencing, cancer treatment, and psychology (Karim, Cochez, Beyan, Decker, & Lange, 2019; Toh & Brody, 2021). Especially in the analysis of one specific type of cancer, SHAP values have been found helpful as they can help to understand interaction effects in some instances (Behravan, Hartikainen, Tengström, Kosma, & Mannermaa, 2020). The study of Richard Du et al. found the application of SHAP values helpful in understanding the prediction of early progression of nonmetastatic nasopharyngeal carcinoma after intensity modulation therapy (Du et al., 2019). The success of their study in applying SHAP values successfully in the field of radiomics gives reason to believe this could also be beneficial for SOMA. However, as SOMA is applied to a very heterogeneous group of lesions, the complexity of the problem might be higher.

Unfortunately, the relative novelty of applying SHAP values in a machine learning context does not allow for an extensive amount of research on them in the field of radiomics. However, further contributions are required to identify the full extent of the capabilities and limitations of SHAP values.

3. Materials

The underlying data used in this thesis is collected from patients who have been treated at the Champaulimaud Institute in Lisbon, Portugal. In total, 634 lesions from 174 patients were treated with SOMA over nine years, starting in 2011. This results in two distinct sets of data. Firstly, aggregated on the patient level, second, one with each metastasis's treatment parameters.

In this thesis, both data sets are treated separately, as they offer different opportunities to generate insights, despite their overlapping nature. Analysis on the patient's level potentially unveils how the treatment affects different types of cancer, cancer location, or patient parameters. This could generate insights into which patients are more susceptible to this specific treatment and which might not be. On the other hand, data on each lesion treatment could offer insights into how well the treatment performs on tumors in various locations, sizes, or metastatic behaviour.

3.1 Data Description

The six dependent variables predicted in this thesis are the following:

Two of the dependent variables are on the lesion level. For this, only data until a post-radiation-therapy measurement is taken into consideration. Local relapse of a lesion is forecasted as a binary classification problem. Furthermore, the local relapse-free survival in months (LRFS), so the time until a lesion reoccurred is forecasted as part of a regression problem.

Four of the dependent variables are on the patient level. Aggregated data of the treatments and follow-up examinations are taken into account. Two regression and classification problems are examined for this dataset. The first dependent regression variable describes the time that a patient survived after the first treatment, in months. The second describes the time in months until the patient obtained polymetastatic status. This means a patient has five or more metastasis simultaneously, which means SOMA is no further applied as it requires oligometastatic status (four or less lesions). The two classification variables describe whether the patient developed more than ten lesions over the treatment period and whether the cancer became polymetastatic.

The independent variables describe the patient, the tumor and metastasis in size, activity and location, the frequency and type of treatment methods, and radiation therapy-derived measurements. A complete list of variables and their description is appended.

Difficulties in obtaining medical data like this are numerous. One of them is the limited number of patients treated with this treatment. Additionally, to have consistent data, the machines need to be the same and identically calibrated to derive the same measurements. This makes it practically impossible to aggregate data from several machines consistently. Finally, the data needs to be collected over many years, and constant follow-up checks need to be scheduled with the patients to obtain the most recent information from them. This proves to be a specifically vulnerable point in the procedure of collecting the data. If a treated patient decides

not to follow up anymore, his data becomes potentially incomplete or, in the worst-case, wrong. Furthermore, recently treated patients cannot be included, as not enough time passed to reason about the effectiveness of the treatment.

4. Methods

This thesis aims to assess if machine learning methods can deliver decision support for SOMA therapy. On the one hand, this includes finding well-performing machine learning methods and evaluating them on their predictive power. On the other hand, the understanding of these methods needs to be fostered. The doctors, who ultimately have to treat a patient, should trust and embrace the information obtained from the machine learning methods. This requires an explanation about a model's functionality beyond prediction accuracy scores. To achieve this, the machine learning models are benchmarked against each other across various error metrics, and SHAP values are examined to investigate coherence with medical intuition and research.

The following chapter will guide through the machine learning methods used to obtain the results of this thesis. After an overview, the underlying methods, their origin, strengths and weaknesses, and their technical implementations are described in detail. Afterwards, the experimental setup to apply the methods to the data is outlined.

4.1 Feature selection

Generally, in radiomics, feature selection plays a significant role due to the numerous features (Parmar, Grossmann, Bussink, Lambin, & Aerts, 2015) derived from radiological images. In this study, the doctors provided a preselection of features for analysis. This already reduces the number of features significantly, and equally, the influence of more minor relevant features. Nevertheless, feature selection remains crucial to the algorithm's performance to prevent irrelevant or redundant data from influencing the algorithm's training and avoid overfitting.

For feature selection, Recursive Feature Elimination (RFE), a wrapper-based method, was applied. As the scope of this thesis is instead to compare the performance of the algorithms and not to explore ideal machine learning pipelines, this method was deemed sufficient to serve the purpose of feature selection.

Recursive Feature Elimination (RFE) was initially introduced by Kohavi (Kohavi & John, 1997). A wrapper-based feature selector utilizes an induction algorithm to find a good subset of features as part of the evaluation function. Initially, RFE fits the entire data set and scores each feature according to the importance. The least important feature is then eliminated, and the remaining features are fitted again to the model. This process is repeated until the desired number of features remains. The model opted for in the context of this thesis is the Decision Tree Algorithm.

A weakness of the algorithm is that the desired number of features needs to be determined beforehand. To mitigate this problem and provide every algorithm with the best possible options to perform well, the option to select the four, seven, or twelve most important features was provided. These are not arbitrary numbers but follow a beforehand conducted exploration with different algorithms. However, with the proposed methodology, every additional option in features would double the computational effort. Hence, limitations had to be made. The python

implementation used to generate the results shown in this thesis is part of the scikit-learn library.

4.2 Handling imbalanced data

Class imbalance describes a problem in classification tasks when there are unequal numbers of samples for each class in the data. Class imbalance poses a great challenge for machine learning algorithms as it injects bias into the algorithm's learning if not accounted for. While altering and adjusting the classification algorithm itself is among the possible options, more commonly used is the resampling of the underlying data. Removing individual instances of the majority class is generally referred to as undersampling. Creating artificial instances close to the samples of the minority class is referred to as oversampling. A wide range of algorithms have been introduced for data resampling with different degrees of randomness or level of information to select instances to sample from.

For the choice of an appropriate resampling method, the shape and the volume of the data should be considered. In cases of very few instances, undersampling would even further reduce the instances for the algorithm to learn from, hence make the problem potentially more challenging. On the other hand, if there is a sufficient amount of data in the minority class, oversampling would introduce unnecessarily artificial data. In this scenario, eliminating some instances of the majority class would not harm the algorithm's performance.

The SMOTE (Synthetic Minority Over-sampling Technique) algorithm was introduced in the Journal of artificial intelligence in 2002 by Chawla et al. (Chawla, 2002). Chawla found that the SMOTE algorithm would perform better than random duplication of minority class instances by reducing the overfitting behavior (Chawla, 2002). It does so by creating an artificial instance in between two randomly selected minority class neighbors.

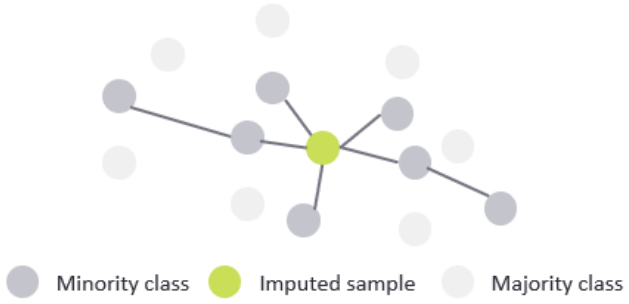

Figure 1 - A two dimensional schema of the imputation of a sample by the SMOTE algorithm. In this examlple seven minority class samples are used to impute one sample.

The new instance($\vec{x}$) is created as follows: An instance from the minority class($\vec{a}$) is selected randomly. Among *k* class neighbors, another instance ($\vec{b}$) is selected at random with *(w),* a random weight w ~ U[0,1]. The new instance is linearly interpolated as:

$$\vec{x} = \vec{a} + w \times (\vec{b} - \vec{a})$$

While the SMOTE algorithm is elegantly simple, it has two disadvantages. Firstly, the algorithm is highly random in picking the samples and neighbors, ignoring possible structure within the data. In the worst case, this can lead to noise amplification within the data. As all samples of the minority class are picked with uniform probability, those minority samples similar to the majority class can introduce more noise into the data when selected (Bunkhumpornpat, 2009).

Secondly, the algorithm does not distinguish between instances in overlapping areas or instances in clearly separated areas of the classes. This leads to the potential introduction of additional noise from selecting or creating instances outside the optimal decision boundary. Hence, artificial instances can be created as instances similar to those from the majority class rather than those from the minority class (Prati, 2004).

Despite its drawbacks, SMOTE remains one of the most well-known oversampling techniques due to its simplicity. Further adaptations were proposed following its introduction in 2002 to address its weaknesses, and some will be discussed below.

As a development of the original SMOTE algorithm (Han & Mao, 2005), Han et al. proposed the decision boundary enforcing algorithm Borderline-SMOTE. It aims to reduce noise by improving one of the weaknesses of the SMOTE algorithm by altering the random selection process. The Borderline-SMOTE algorithm selects instances at the class border or close to it, and the labels of the k-nearest neighbors are the decision criteria for whether it is identified as noise or not. Additionally, these labels are also the criteria for whether an instance is close enough to the border for interpolation or too far away and not selected for interpolation.

As displayed in Figure 2, the Borderline-SMOTE algorithm selects samples from the given, unaltered data (left figure) from the minority class close to the class border and interpolates them (right figure).

Figure 2 - A two dimensional schema of the Borderline-SMOTE algorithm. The imputed samples are imputed between the points closest to the boundary between minority and majority class.

SMOTE-Tomek is another adaptation of the SMOTE algorithm that combines over- and undersampling. After oversampling with the basic SMOTE algorithm, as discussed above, Tomek links (Tomek, 1976) are used to create more distinct clusters. A Tomek link is present when two points from different classes have the smallest distance to each other than to any other point to

Figure 3 - A two dimensional schema of SMOTE-Tomek. Majority samples within the space of the minority samples are filtered out after samples are imputed in this space.

either of them. If two instances create a Tomek link after the SMOTE algorithm was applied, then one of these two is discarded as noise, or both are discarded as borderline instances.

A weakness of the SMOTE-Tomek adaptation is the possible removal of minority class instances to improve the decision boundary between the classes. Furthermore, as noise is removed as it is part of a Tomek link, information is lost. Nevertheless, SMOTE-Tomek has been proved to be effective in some cases to obtain better results compared to other over- and undersampling methods (Batista, 2004).

4.3 Machine learning algorithms

To effectively compare the performance of different models in the context of this thesis, models with various underlying mechanisms and theoretical concepts were chosen. However, when comparing Machine Learning models, one should keep in mind that the conclusions drawn from their comparison is depending on the underlying data and not globally applicable to all problems. This does mean that the "No Free Lunch Theorem for Search Algorithms" (Wolpert & Macready, 1996) does hold. It states that over all possible problems no algorithm is superior to any other algorithm. However, over one problem, some algorithms may perform better than others due to the shape of the underlying data (Dietterich). The difference in results is usually a function of noise, variance, and bias in the data, which leads to error that can not be mitigated with the algorithm's capabilities.

In the following section, the different algorithms will be compared in their underlying function.

Linear Models

Lasso Regression

Lasso regression (Lasso) is an extension of a linear regression model that can exclude variables through L1 Regularization. Lasso refers to the regularization and is an abbreviation for the Least Absolute Shrinkage and Selection Operator. Lasso Regression has been originally applied in the geophysics literature in 1986 (Santosa & Symes, 1986). However, it is based on decades of previous work in statistics.

Like in linear regression, the goal of the Lasso is to fit a line that best describes the data by minizmizing the error between the predicted valies and the drue dependent variable values. Lasso regression includes a penalty term, the L1 regularization. This penalization enforces the L1-norm of the fitted weight coefficients to be low and is directed towards independent variables that do not significantly influence the dependent variable. Their influence on the model is reduced to zero.

Furthermore, it utilizes a technique called shrinkage. Shrinkage is favorable in simple and sparse data models because values are being shrunk towards a central point, such as the mean. This should lead to a more significant decision boundary.

The addition of regularization has one distinct advantage over the linear regression that it makes the model more robust to overfitting on the training data. While reducing the number of features makes the model more robust, it also makes it more interpretable from the values of its coefficients.

The implementation of the Lasso Regression, which was used to generate the results of this thesis, is part of the scikit-learn library. The underlying optimization algorithm to fit the model to the data is coordinate descent (a method similar to gradient descent). The strength of the regularization can be determined in the hyperparameter settings and the convergence behavior

of the optimization algorithm, allowing for various degrees of randomness in the search for optimal solutions.

Logistic Regression

Despite the misleading name, Logistic Regression (Grosu et al.) is a classification model. In statistics, it is also referred to as the logit – or logistic model. The initial development as a statistical model in 1944 is ascribed to Joseph Cramer (Cramer, 2002) despite many others contributing to it as early as in the 19th century. In its basic form, the logistic regression is only able to model a binary dependent variable.

Similar to linear regression, the logistic regression tries to fit a line to the underlying data. The shape of the line is 'S' shaped and called a sigmoid function. The sigmoid function follows the formula:

$$sigmoid = \frac{1}{1 + e^{-x}}$$

X is the weighted sum of all input features of one sample. The function returns a value between zero and one to classify data belonging into two different categories.

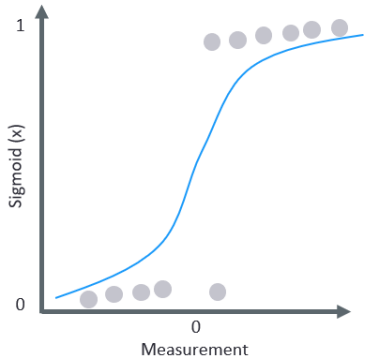

Figure 4 - The logistic regression algorithm seperating samples according to the best fit of the sigmoid function

Since LR poses a non-convex optimization problem, one needs to fit the sigmoid function iteratively using, e.g. Gradient Descent.

Logistic regression is considered one of the simplest classifiers in machine learning. It is easy to interpret and provides the feature importance as its weighting coefficients by default. With its regularization ability, it provides a measure to counteract overfitting behavior by eliminating less essential features. However, the assumption of linear separability between the independent- and dependent variables and the inability to solve non-linear problems are significant limitations.

The python implementation used to generate the results shown in this thesis is part of the scikit-learn library. The implementation comes with various optimization algorithm implementations, L1 and L2 regularization options to mitigate the influence of unimportant features, and the possibility of adapting to multiclass classification.

Non-linear Models

Classification and regression trees (CART)

Classification and regression trees, also called decision trees (DT / DTC/ DTR), are amongst the easiest to understand, best interpretable, and visually self-explanatory algorithms of supervised machine learning. While many researchers contributed to developing different tree-based algorithms, among the most influential contributions in the machine learning community is the work on 'Classification and Regression trees' by Leo Breiman (L. e. a. Breiman, 1998). Especially in a non-machine learning or non-statistical context, decision trees can bridge the gap between a well-performing model and understanding of and trust in the model.

A decision tree consists of branches, nodes, and leaves. Every branch consists of several nodes, splitting the data into two - not necessarily even – parts. The terminals of each branch are called leaves. The decision tree splits data at every node of the tree until a convergence criterion is met, or the data cannot be split any further. A variety of metrics determines the calculation of each splitting point on the data. In the case of classification, the Gini index or Entropy are common metrics. For regression trees, the residual or the mean squared error serves as the most common metrics (L. e. a. Breiman, 1998).

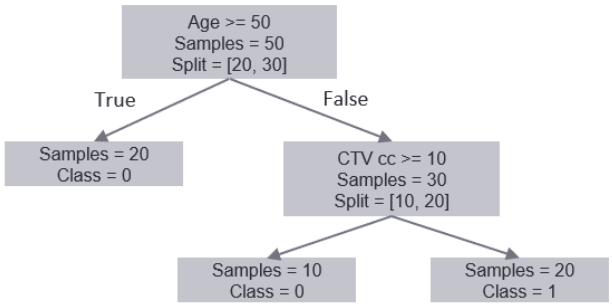

Figure 5 - schema of a decision tree with two nodes and tree leaves

The values obtained from each leaf of a regression tree are the average of the training sample observation residing in this node to obtain the predictions. In classification cases, the value obtained from each leaf is the mode (class) of the training sample observations residing in this node.

The key advantages of decision trees are their inbuilt feature selection mechanism, straightforward interpretation, and visualization. Key disadvantages are their overfitting behavior, especially on small datasets, low variance data, and vulnerability to unbalanced data. Nevertheless, the tree can mitigate some of these challenges by limiting depth or the number of nodes (called pruning) or grouping many trees in an ensemble.

The implementation of the decision tree algorithm used in this thesis to derive the presented results is part of the python library scikit-learn. The implementation allows for various parameters that influence the final structure of the tree, such as the number of terminal nodes, number of leaves, or the minimum number of samples within each leaf. Furthermore, it allows for various pruning parameters, as well as the above described split measures.

K-Nearest Neighbours

The K-Nearest Neighbors algorithm (KNN) is a classification and regression algorithm. It was first developed by Evelyn Fix and Joseph Hodges in 1951(Fix & Hodges, 1951). The algorithm is based on the idea that a number (k) of closest samples to one object have predictive power over the object.

The algorithm works slightly differently for regression and classification problems. In both cases, the algorithm first translates the data into vectors. Then it calculates the distance - usually the Euclidean distance - of each vector to the test data vectors. The Euclidean distance 'd' is calculated with the following formula, where 'p' and 'q' are the two points and 'n' the number of features or dimensionality of the vector.

$$d(p,q) = \sqrt{\sum_{i=1}^{n}(q_i - p_i)^2}$$

In a classification problem, the k closest neighbors majority class is the predicted class. In a regression problem, the average vector of the k nearest neighbors is the predicted value. Alterations of the original algorithm allow for different weights of each neighbor relative to their distance and different distance measures, such as the Minkowski distance or Manhattan distance.

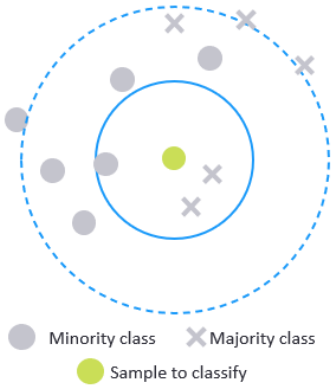

Figure 6 - schema of the KNN algorithmwith two numbers of (k) [2,5] leading to different classifications of the sample to be classified

KNN is a non-parametric algorithm that assumes a spherical gaussian distribution of the data. Furthermore, the training process is straightforward and brief. However, finding the correct number of k is difficult. Grid-searches can mitigate this weakness. Additionally, data points located on a decision boundary are handled poorly by the algorithm.

KNN is often used for identifying groups of data within a dataset. In the context of the thesis, this could be beneficial, as patients with a similar medical history might benefit from similar treatment. On the other hand, the complexity of understanding the medical implications of patients' data might be beyond the proximity of data, which would make it difficult for KNN to identify these.

The python implementation of this algorithm's regressor and classifier is part of the scikit-learn library. It allows all previously mentioned parameters as hyper-parameters. Per default, it calculates the distance with the Minkowski distance, which is the sum of the absolute distance between two points.

Gradient Descent

Gradient Descent is a flexible optimization technique and not able to derive predictions by itself. Cauchy first suggested gradient Descent in 1847 (Lemaréchal, 2012). However, for non-linear optimization problems, it was first studied by Haskell Curry in 1944 (Curry, 1944).

Gradient Descent optimizes a model, e.g., logistic regression, by finding a (local) optimum to its loss function. Starting at a random point, the underlying idea of the algorithm is to step along the loss functions' gradient at the current point in the opposite direction until a convergence criterion is reached. The differentiable function is derived from the error of the underlying model. The learning rate regulates the size of each step taken along the gradient of the differentiable function, and the number of steps is finite. In this manner, the error is reduced until a minimum is reached, and the gradient is zero. This procedure is greedy, as it always follows the direction of the steepest descent. In non-convex loss landscapes, this might not lead to the global minimum, the optimal solution.

Stochastic Gradient Descent

Stochastic Gradient Decent (SGD) is a development of the gradient descent algorithm. Instead of evaluating the full gradient for each training data sample, it approximates the gradient using a random subset of the data. This stochastic approximation can be traced back to the Robbins-Monro algorithm (Robbins & Monro, 1951).

The stochastic approximation allows the algorithm to perform on large datasets as it cuts the high computational load that would be otherwise associated with the gradient calculation. This leads to a faster conversion behavior of the algorithm. Smaller steps can converge slower with a lower learning rate, allowing for better approximation in most cases.

The python implementation of the algorithm, which was used to generate the results of this thesis, is part of the scikit-learn library. It offers a classifier and regressor. Many different loss function approaches, such as SVMs, linear and logistic regressions, or neural networks, can be selected. Furthermore, they can be regularized by the L1 and L2 regularizers for feature selection. Furthermore, stopping criteria, learning rate, number of iterations, schedules for the learning rate, and other parameters can be customized. This makes this technique very versatile in its application and an exciting addition to this thesis.

Ensemble Methods

Bagging

Ensemble methods combine various base models to form a better predictive model than each base model would be on its own. The first to develop this method was Leo Breiman in 1996 with the bootstrapping aggregating method, in short, Bagging (Leo

Breiman, 1996). In this method, bootstrap samples are generated by subsampling the data (uniformly subsampling with replacement). Individual models train on one bootstrap and their results are aggregated as a majority vote in classification or averaging in regression. This technique helps to reduce variance and overfitting of the model.

Various algorithms were developed on this basic principle: Combine various individual models, trained on subsamples of the data, and aggregate their results. Some of these algorithms are now described as they were applied in the scope of this thesis

Random Forest

The Random Forest (RF) algorithm is one of the most frequently used algorithms of the ensemble methods family. Leo Breiman introduced random Forests in 2001 (Leo

Breiman, 2001). The underlying principle of this algorithm is that many uncorrelated trees trained on different subsamples of the data vote for the outcome of a prediction. Low correlation between the individual trees is essential to reap the benefits of the algorithm. Each error of an individual tree should be overruled by the ensemble as long as not all errors are similarly directed.

Technically the algorithm combines several decision tree predictors to one ensemble model. Each of these trees is trained on a different subsample (with replacement) of the data and a randomly selected set of features (with replacement) of the training data. Predictions are formulated by the vote of each tree in a classification problem or the average of each tree in a regression problem.

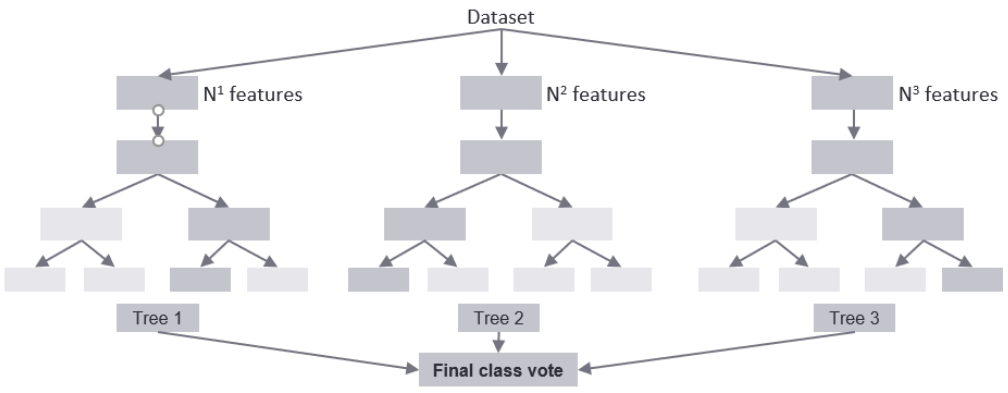

Figure 7 - schema of the random forest algorithms. three subsets of the data train individual trees, that conclude one final class vote

It is essential to make sure that the features in the training data need to have at least some predictive power over the prediction target. Features without predictive power will not be excluded by the algorithm and will introduce bias with wrong predicting trees. Furthermore, the number of subsamples and features that each tree is trained on will influence the uncorrelatedness of the trees. Trees trained on the same data will have a very high correlation. Hence, they will always come to similar or identical predictions.

The python implementation of the random forest algorithm, which was used to generate the results of this thesis, is part of the scikit-learn library. The implementation is based on the work of (Leo

Breiman, 2001). Both the regressor and classifier offer a variety of parameters to optimize the performance. They range from the number of trees, the size of each tree, the various split criteria and limitations of the tree branches, the number of features to train every tree, and other parameters.

Boosting

Boosting is another machine learning technique developed by Leo Breiman (Leo Breiman, 1997). The underlying principle of boosting is to have a series of predictors where each predictor learns from the previous predictor in the series. Various machine learning algorithms, such as tree-based models, gradient descent, or regressors, can be boosted. As boosted models learn from previously committed mistakes, it should take, in theory, less time to come to close to optimal solutions. However, the stopping criteria must be defined well not to stop before closing in on optimal solutions or overfitting the model.

Adaptive Boosting

Adaptive Boosting (Ada Boost) was first introduced in 1996 by Freund and Shapire (Freund & Schapire, 1996). It is considered well-performing without extensive hyperparameter optimization due to its properties (Kégl, 2013). Therefore, overfitting can be less problematic while training close to optimal models.

The underlying principle of the algorithm is that several algorithms – so-called 'weak learners' - are combined and weighted to form the output of the model. A weak learner is a model that performs better than guessing but is far away from the optimal solution. The weak learners subsequently learn from the errors of the preceding weak learners.

To evaluate the performance of each weak learner, more weight is assigned to incorrectly classified samples in the case of classification or high error samples in the case of regression. This helps to foster iterative learning behavior. Every weak learner is also evaluated against the other weak learners to give better performing weak learners a more significant impact on the overall output.

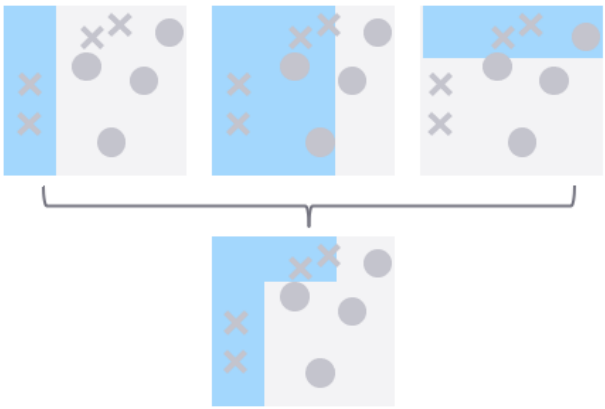

Figure 8 - schema of three boosting stages

The Ada Boost algorithm utilizes decision tree stumps as weak learners with one tree node that splits into two leaves. In combination with the weighting behavior of the algorithm explained above, the algorithm automatically selects the essential features by itself when being trained. However, the progressive learning behavior has a downside: it is susceptible to learning noise and outliers.

The python implementation for the classifier and regressor of the Ada Boost algorithm, which was used to generate the results of this thesis, is part of the scikit-learn package. The implementation is based on Freund and Schapires' work (Freund & Schapire, 1997). The hyper-parameters to optimize the algorithm are limited to the number of weak learners, the learning rate, the loss function in the case of regression, and the weighting algorithm option in the case of classification.

Extreme Gradient Boosting

Extreme Gradient Boost (XGBoost) is a relatively new machine learning method introduced in 2016 Chen and Guestrin (Chen & Guestrin, 2016). The machine learning community considers it among the highest performing algorithm that has won numerous competitions and is still optimized by over 350 collaborators.

XGBoost is part of the ensemble tree algorithm family, like the Ada Boost algorithm. However, contrary to the Ada Boost algorithm, it utilizes a different boosting strategy. XGBoost uses the Gradient Decent algorithm to minimize the errors to produce superior results with below-average computing resources. Hence, it is a development of the Gradient Boosting algorithm introduced by Friedman (Friedman, 2001).

The Gradient Boosting algorithm, just as XGBoost and the Ada Boost algorithm, is an ensemble of weak learners, typically decision trees. In Gradient Boosting and XGBoost, a loss function is defined and optimized using gradient descent. Predictions are being updated utilizing a learning rate to find the optimal loss function error. The intuition of Gradient boosting is to detect patterns in the errors and subsequently eliminate these errors by modeling them in its' weak learners.

XGBoost has three additional capabilities :

1. Awareness of data sparsity – sparse features, such as one-hot encoded categorical variables or missing data, by learning from the training loss. Therefore, it can handle different types of sparse patterns.
2. Regularization of complex models – regularization penalizes the complexity of the model, thus preventing overfitting. Lasso (L1) and Ridge (L2) minimize or nullify the impact of low-importance features to reduce the complexity of the model.
3. Weighted data set handling - XGB utilizes the weighted Quantile Sketch algorithm (Chen & Guestrin, 2016). This results in better tree node splitting decisions in weighted datasets.

The large amount of hyper-parameters available for XGBoost allows for extensive customization of the model to the underlying data. Tree depth and number of nodes, the learning rate, regularization, number of weak learners, to name a few, provide a large number of options to influence the outcome of the model positively. With extensive customizability comes the downside of possible overfitting behavior, which should be accounted for, potentially with regularization.

The python implementation is based on the work of Chen and Guestrin. As it is a community-developed algorithm that is still being advanced, it is not part of the scikit-learn library but available in a public repository (https://github.com/dmlc/xgboost).

Support Vector Machines

Support Vector Machines (SVMs) is a supervised machine learning method for classification and regression problems. Vapnik and Chervonenkis originally developed the model in 1963. Vapnik continued development until first publication in 1995 (Boser, Guyon, & Vapnik, 1992) (Cortes & Vapnik, 1995). The underlying VC Theory of Vapnik and Chervonenkis serves as its statistical framework. The VC Theory attempts to describe a statistical approach to how computational learning can be conducted.

For classification problems, SVMs construct several maximum-margin-hyperplanes which linearly separate data points in a high or infinite-dimensional space. Maximum-margin-hyperplanes try to maximize the distance between two data points of separate classes and splitting them with a hyperplane to serve as a decision boundary. The more distance the SVM can bridge with the hyperplane that separates the two classes, the better the generalization ability of the algorithm. In the first introduction of the algorithm, only linearly separable data could be successfully split.

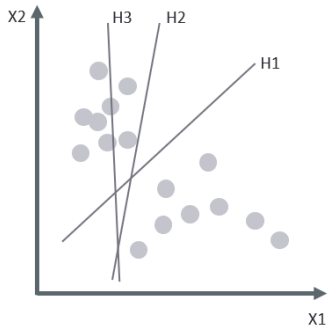

Figure 9 - schema of the SVM. Three hyperplanes seperate the data differently, where H1 is favorable, as it has the largest distance to both classes

With the later introduced 'Kernel Trick' and application of 'Soft Margins', this limitation was mitigated. The Kernel trick projects the data into a higher, potentially infinite, dimensional space to assume linear separability statistically. This allows separating nonlinear separated data points to be separated linearly. Different kernels transform the data according to varying shapes. Hence, the shape of the initial data will influence how well different kernels can separate the data.

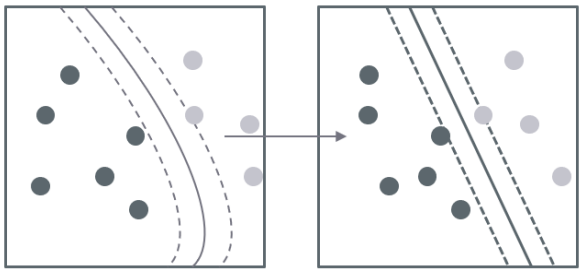

Figure 10 - schema of the kernel trick, transforming the data into a higher dimensional space to ensure linear separability.

Applying 'Soft Margins' to the SVM allows the algorithm to tolerate misclassifications of the data. The degree of misclassification that will be tolerated can be specified through hyper-parameters for any problem that is not linearly separable and for those who are not, even in a higher-dimensional space.

For regression problems, an abbreviation of the classification algorithm was introduced in 1997 (Drucker, Burges, Kaufman, Smola, & Vapnik, 1997). To cope with a regression task's increased complexity, a different loss function is introduced with an insensitive loss term ε. The insensitive loss term constructs a virtual tube around the hyperplane with the radius of the loss term ε. All observed values within this tube are not penalized. However, values outside the tube are. Additionally, slack variables can be added to allow for additional errors and approximation.

The technical python implementation, which was used to generate the results of this thesis, is part of the scikit-learn library. The regression and the classification algorithm are based on the work of LIBSVM by Chang (Chang & Lin, 2011). In addition to the hyper-parameters above, the scikit-learn library offers various other parameters to optimize the algorithms' performance.

Genetic Programming

Genetic Programming is a branch of Genetic Algorithms with the same fundamental characteristics but a different representation of a solution. Genetic Algorithm (GA) is an evolution-based search and optimization algorithm derived from Charles Darwins' biological evolution theory (Goldberg, 1989). The essence of this algorithm is that a population (of possible solutions) will evolve (or improve) over time by selection and variation of the individuals.

There are numerous variations of the Genetic Algorithm. However, they share the same elements: a population of potential solutions, a selection algorithm that selects individuals from the population based on a fitness score for each solution, and variation introduced into the population via crossover or mutation. While mutation alters an individual randomly, crossover utilizes the variation inside the population and exchanges information between individuals.

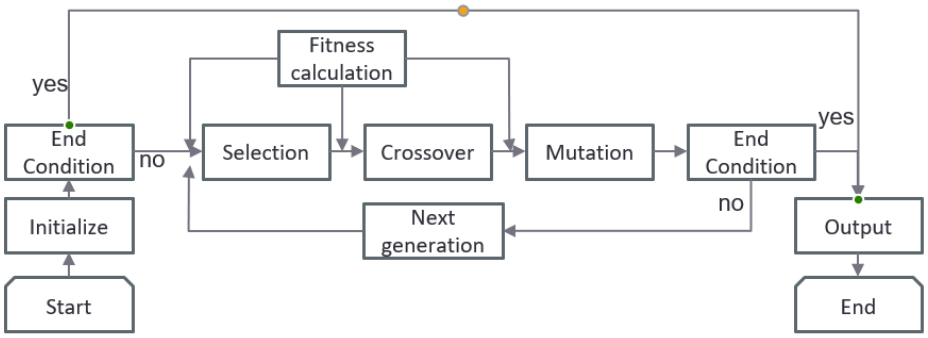

Figure 11 - schema of the genetic algorithms' process

Genetic Algorithms evolve in cycles to find the best possible solution. Each cycle begins with an initial population of solutions. The selection algorithm chooses the most suitable individual solutions that should evolve into the population's subsequent population. To find the individuals that the algorithm deems as most suitable, a 'fitness function' formulated in the search algorithm to evaluate each individuals fitness. The selected individuals are then altered by selecting numerous crossover and mutation methods before they complete the cycle and represent the subsequent population. This cycle is repeated until a predefined stopping criterion is met. While Genetic Algorithms represent a solution as a string of numbers, Genetic Programming solutions are computer programs in lisp or scheme computer languages as described by Koza (Koza, 1994).

To optimize for the best configuration of the algorithm, specific parameters should be taken into account. First, larger population size and more evolution cycles increase the possibility for finding reasonable solutions at the risk of overfitting. Secondly, the selection algorithm has to balance between selecting the best individuals and maintaining various individuals in the population. More greedy selection criteria prefer reasonable solutions, thus, sacrificing variance in the population. This leads to faster conversion of the population around a specific solution, which is not necessarily the best possible solution. Thirdly, crossover and mutation inject variance into the population. While crossover utilizes the information of at least two individuals of the population to create new individuals, mutation alters one individual at a random rate. This implies that mutation is more invasive than a crossover, meaning that an individual is altered at a higher rate than its previous individual(s). A probability to crossover and mutation is assigned at which rate either method is applied to individuals of the population.

The python implementation used to obtain the results in this thesis is part of the GP-learn library. All parameters mentioned previously can be optimized for in this implementation. The high computational effort required the application of a smaller hyperparameter grid.

Artificial Neural Networks

As the name suggests, artificial neural networks try to artificially imitate the learning and decision process – in other words, the intelligence – of a network of neurons. In essence, the human brain. The rationale behind this technique is that a machine provided with the same capabilities and information as the human brain can imitate the human brain's learning process. However, machines have the advantage of processing data much faster and more accurately than the human brain, which should make them superior at specific tasks.

There are three main architectures of neural networks, the Artificial Neural Network (ANN), the Convolutional Neural Network (Tomek, 1976), and the Recurring Neural Network (RNN). Each of them is serving a different purpose. In the context of this thesis, an Artificial Neural Network is used, as it is the best suited for the research task.

The first computational model for a neural network dates back to 1943 (McCulloch & Pitts, 1943), while the research in understanding neural interaction started as early as the late 19th century. In 1949 Hebb (Hebb, 1949) introduced 'Neural Plasticity', a concept that states that neural connections are non-static, inferring that strengthening neural connection is what learning means in an anatomical sense. Based on these principles Rosenblatt developed the single-layer perceptron (Rosenblatt, 1962), a cornerstone of the current understanding of neural networks. After the extension to multiple-layer perceptron networks (MLP), another cornerstone of the current understanding was formulated by Werbos in 1975 with the concept of backpropagation (Werbos, 1975).

An MLP neural network consists of several layers of neurons. An input layer, an output layer, and any number of so-called hidden layers in between.

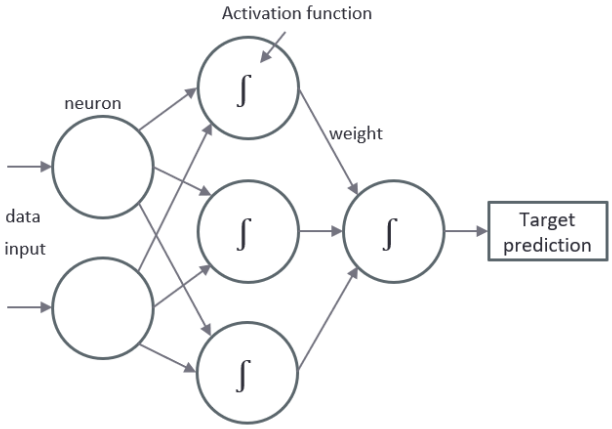

Figure 12 - simplified schema of an ANN with an input layer, one hidden layer and a output neuron

Every neuron of the network follows the same mathematical principle. It takes the sum-product of input (the input data for the input layer or output of a previous layer) and weight for this specific input. In addition to the sum-product, a bias constant might be added before an activation function is applied. The activation function, often the sigmoid function, converts the value to a specific range, thus, all outputs fall into this range. In the case of the sigmoid function, the output is between zero and one.

$$sigmoid = \frac{1}{1 + e^{-x}}$$

As all neurons are connected to every neuron of the previous and subsequent layer, the weights processing the inputs determine the neural network's performance. To improve the weights, for the neural network to model the input correctly, different optimization algorithms, such as gradient descent, can be applied. Alternatively, backpropagation can alter the weights, which influenced a faulty output in the previous training process.

The most significant advantage of neural networks is that they can learn complex relationships in the data that escape the human eye or mind. This includes learning from non-exhaustive data and having an error tolerance. Furthermore, the information is stored in the weights of the neural network. However, neural networks often require a large amount of data to adjust weights correctly. Recent studies on very deep architectures, tools, and transfer learning on minimal data sets have shown potential. However, they come with much complexity (Pasini, 2015a) (Pasupa & Sunhem, 2016).

The python implementation used to derive the results discussed in this thesis is part of the python library scikit-learn. It includes an MLP classifier and regressor without backpropagation. However, regularization options, the architecture, and different optimization algorithms and parameters provide numerous possibilities to optimize the model to the underlying data.

4.4 Error measures

In this section, the measures on which the performance for the different problems is evaluated are discussed. When trying to evaluate the algorithms' performance, classification and regression different measures of success are required. Additionally, there is no single correct measure to evaluate the performance of an algorithm, but the correct measures must be selected in the work context. Especially in the medical context, when dealing with patient's data, the error measure should be profound. Furthermore, for classification problems, class imbalance must be considered, as it influences the importance of errors certain.

In a classification, assessment metrics can be derived from the confusion matrix. The confusion matrix can be constructed from the predictions, categorizing them into true and false predictions for each of their classes (Nathalie Japkowicz). The absolute values of the confusion matrix give the full context of the classifier's performance.

	Positives	Negatives	
Predicted positives	True positives	False positives	Precision $\frac{TP}{PP}$
Predicted negatives	False negatives	True negatives	
	Recall $\frac{TP}{P}$	Specificity $\frac{TN}{N}$	

Figure 13 - Confusion matrix

In classification problems, the accuracy rate of the algorithm describes the percentage of correctly classified samples, divided by the total number of samples. On the one hand, this measure is straightforward to understand and unambiguous. On the other hand, it loses its meaningfulness in imbalanced datasets because it will always bias the majority class. Suppose one class is significantly underrepresented in the data. In that case, the algorithm might classify all instances as the majority class, and only those few of the minority class will be classified wrong. However, the accuracy will not represent the performance accurately. Therefore, further measures should be evaluated.

Recall (also referred to as sensitivity or true positive rate) displays the accuracy within the positives. The specificity is the pendant for the negative class. The precision is the accuracy of the predicted positives. The F1-score combines recall and precision in a weighted average. Although less intuitive, the F1 score is generally the better error measure in class imbalance compared to the accuracy.

$$F1 - score = \frac{2 * (Precision * Recall)}{(Precision + Recall}$$

In imbalanced classification problems, it is essential to evaluate the algorithm's performance on classifying the minority class samples combined with the accuracy rate. Especially in a medical context, this metric is essential as the minority class can often be of particular interest.

In regression problems, the performance is evaluated based on four error metrics: the mean absolute error, the mean squared error, the root mean squared error, and the R^2, also called the coefficient of determination. The mean absolute error and mean squared error are the absolute or mean difference between the predicted and actual values. The mean squared error is the square root of the mean squared error. R^2 compares the models' performance with a constant baseline, the mean value of the observations. The scores are always smaller than one.

$$R^2 = 1 - \frac{MSE\ (model)}{MSE\ (baseline)}$$

The scores of these evaluation measures of each best model configuration are averaged over 30 different initiations (seeds) of the same model configuration. These 30 initiations are carried out to split the training and test data sets several times to observe the spread of results to find robust models, especially on small data sets.

4.5 Statistical evaluation methods

To objectively compare the performance of the machine learning algorithms, the Wilcoxon signed-rank test is used (Wilcoxon, 1945). It is a non-parametric statistical hypothesis test that compares two related samples on a statistically significant difference in mean ranks. The test does not have assumptions about the distribution of the samples. In the context of this thesis, it is the best suited statistical test because error scores of two algorithms from 30 different sample combinations of the test and training set are compared. The 30 different sample combinations are identical for both algorithms, hence, enables the comparison in pairs.

The null hypothesis (de Hoon, 2004) and alternative hypothesis (H_1) tested for are as follows:

H_0: the difference between the pairs follows a symmetric distribution around zero
H_1: the difference between the pairs does not follow a symmetric distribution around zero

The resulting W statistics can be compared to critical values of a reference table. The two-sided test rejects the null hypothesis if the absolute W statistic is larger than the critical W value. P-values are obtained to evaluate the statistical significance.

4.6 SHAP Values

SHAP values, or SHapley Additive exPlanations, is a framework introduced in 2017 by Lundberg and Lee (Lundberg & Lee, 2017). It is a framework that aims at interpreting machine learning models of any nature (Lundberg & Lee, 2017). It is a method of additive feature importance, as every single sample contributes to the feature importance of each feature separately and additively. SHAP values are based on a game-theoretical approach from Lloyd Shapley from the 1950s (Shapley, 1953). It is one of his most influential theories contributing to the Nobel prize award in 2012 to Shapley.

The game-theoretical setting of Shapley values is the problem of fairly distributing money between players who contributed to the outcome of a game. Fair in this setting has two properties:

1. The amounts sum up to the amount to be distributed in total.
2. They are consistent with each player's contribution that a player who contributed more always receives more money than a player who contributed less.

The Shapley value for a player i is calculated for game f by an average of the marginal contribution of this player i for every possible subset of players S. The marginal contribution is calculated by the difference in the outcome of the games without and with the player of a specific subset that:

$$f(S \cup \{i\}) - f(S)$$

The differences of each game are averaged for one player to obtain this player's Shapley values (Shapley, 1953). Furthermore, it was proven that Shapley values are the only theoretical method to distribute the money of the game within the constraints of a set of specific desirable properties, always resulting in a fair distribution of money.

The contribution of Lundberg and Lee is to adapt this concept as a machine learning model explanation method where i is each feature contributing to a model prediction f where M are all features included in the model and S, a subset of the features M. Two problems of this method arise in applying Shapley's concept to machine learning that Lundberg and Lee have addressed:

1. A subset of features S will have missing features that have been used by the model that is supposed to be explained.
2. Averaging marginal contribution across all subsets of features for every sample opposes a great computational effort.

Lundberg and Lee applied different explanatory algorithm approaches to overcome these challenges, which can be model-specific or agnostic. Some of the model-specific so-called "explainers", such as the tree-explainer, explaining tree-based algorithms, manage to overcome the two problems stated above. As a missing feature of a tree model results in an undescribed split node in the tree, they interpolate the split from the weighted average of both branches. To overcome the computational effort required, they store some data in the memory to avoid repetitive calculations. The model agnostic algorithm presented by Lee and Lundberg only takes a sampled subset "S" of features into consideration to approximate the SHAP values.

Furthermore, they fill missing values from a background data set that has to be defined by the user. This can make the approximation of the Shapely values less concise. In this thesis, only

these two explainers are used, while other explainers exist for different model types such as linear models or neural networks.

After the Shapley values for each sample are calculated and aggregated, they are aggregated to the feature level and graphically plotted in several ways. Figure14X below illustrates the feature importance of a balanced binary classification problem.

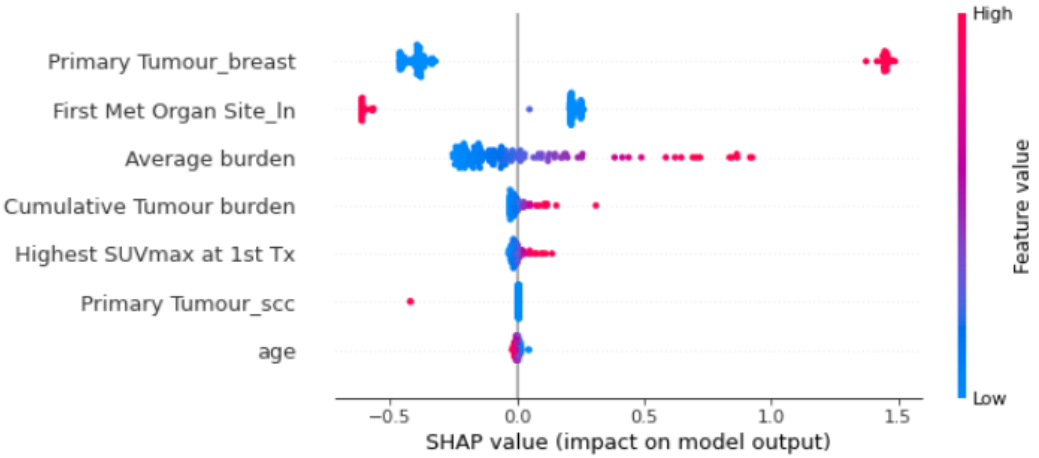

Figure 14 - exemplary display a model interpretation with SHAP values. Feature influence on the model predictive outcome is in decreasing order. Every point is one sample value for each feature. The value itself is displayed by the color.

In Figure 15, every dot in a variable represents a sample and its influence on the SHAP values. The color notes the sample-specific value of the feature. The seven features displayed are in decreasing order of importance, where the exact importance of each feature could be obtained from the underlying aggregate of the SHAP values. This makes the influence of features and each sample on the model's output very visual and easy to understand.

Other additive feature attribution methods try to explain the feature importance, such as LIME (Ribeiro, 2016) for local approximation using regression or DeepLIFT (Shrikumar, Greenside, & Kundaje, 2017) for explaining deep neural networks, have aimed at solving the same problem. However, SHAP values seem to outperform these methods in terms of computational efficiency, consistency with human intuition, and explaining class differences in the initial tests conducted (Lundberg & Lee, 2017).

5. Experimental setup

First, the data is cleaned and pre-processed to provide the algorithms the data in the shape they require while removing noisy or broken samples. After pre-processing, the data is split into a training set, on which a model is trained, and a test set. The test-set remains untouched until performance evaluation. Following the train/test split, the data is scaled. Scaling is necessary, as some algorithms overestimate the impact of features with high absolute values. By scaling the data, this bias can be removed.

Afterward, a feature selection algorithm selects a set of features that best describes the information the machine learning model is trying to learn. For classification tasks, oversampling methods level the classes with equal number of samples between classes. To identify the best practice model for each of the six problems investigated, various algorithms are applied and evaluated against each other. For evaluation, several error metrics are taken into consideration. Finally, the performance of the algorithms are statistically evaluated using the Wilcoxon signed ranked test. SHAP values taken from the best performing model attempt to understand the underlying features that influence the machine learning algorithm's decision.

Each method comes with its complexity and underlying assumptions. To account for that, a flexible environment was provided for each algorithm. This environment allowed to choose between several methods with varying degrees of sophistication and provided each method with a set of four, seven or twelve selected features. For classification tasks, three different oversampling methods are provided to the algorithms.

To find the optimal set of hyperparameters for each algorithm, a brute force approach trying all possible combinations of a set of hyperparameters to find the optimal set was applied. As the amount of data supplied to the algorithms is very limited, it was essential to use cross-validated results to determine the best algorithm configurations. Cross-validation prevents over-fitting of the algorithm to the training data, which leads to poor generalization performance.

After the machine learning model configurations were optimized, the final results are obtained by running the best configurations over 30 seeds. These seeds initiate random parameters in the code differently, resulting in variation in the data split, oversampling methods, and some algorithms' initializations. The variation induced with this technique can be significant because of the datasets' small size. Changes in the samples belonging to the training or test set potentially significantly influence the algorithms' learning ability. Furthermore, 30 sets of results are generated for each algorithm, which allows for better statistical evaluation with the Wilcoxon signed ranked test.

On the best algorithm, the concept of SHAP values was applied to understand how the model derived its predictions from the data. Importantly, SHAP values describe how the input features contribute to a model's conclusions, but they do not imply causality between input features and prediction.

In the next part of this section, the methods applied, their respective strengths, prerequisites, and backgrounds are explained in more detail.

6. Description of results

In the following chapter, the results obtained for the six dependent variables' predictions are described. For every dependent variable, the best result for every algorithm is compared by several evaluation measures. Finally, the SHAP values from the best model are being examined.

Dependent variable: "≤10"

The first variable evaluated is named "≤10" in the data. It is a binary variable, which describes whether a patient developed more than ten oligometastatic lesions throughout his timespan as a patient. The definition of this variable already opposes a difficulty, which is the completeness of the data provided. The data can only display events recorded by the doctors. However, if a patient terminates the doctors' relationship, this data cannot further be obtained. Hence, it might inject noise into the data.

Table 1 describes the performance and configurations of the best parameters of each algorithms' best configurations mean over 30 seeds:

model	sampler	# of features	accuracy training	accuracy test	F1 score	precision	recall
LR	SMOTE	12	0.78	0.74	0.65	0.65	0.66
SVM	SMOTE Borderl.	7	0.7	0.73	0.63	0.64	0.63
SGD	SMOTE	12	0.77	0.73	0.63	0.64	0.66
RF	SMOTE	12	0.9	0.73	0.65	0.64	0.68
GA	SMOTE Tomek	7	0.67	0.73	0.48	0.8	0.35
ADA Boost	SMOTE	12	0.88	0.71	0.61	0.61	0.64
XGBoost	SMOTE	12	1	0.7	0.58	0.59	0.58
MLP	SMOTE	12	0.94	0.7	0.61	0.6	0.64
KNN	SMOTE Borderl.	12	1	0.68	0.56	0.57	0.56
DT	SMOTE Tomek	12	0.81	0.67	0.58	0.56	0.63

Table 1 - results table for the dependent variable "≤10"

All averaged test accuracy scores of the different algorithms range from 74% to 67%. This accuracy score means that the best algorithm fails to predict every fourth patient correctly. The worst algorithm fails to predict every third patient correctly. For most algorithms, the precision and recall scores are very balanced with minor differences. It indicates that the algorithms can learn to distinguish between the two classes evenly, not leaning their prediction towards either of them. This balance is reflected in the F1-score. The most significant difference in the results is the training data accuracy and its' difference to the test data accuracy. The three best algorithms have very similar accuracy on the training and test data, indicating that the model is learning from the data and not remembering each case. Hence, indicating little over-or underfitting behavior, which would need to be further validated. For most other algorithms,

assertive overfitting behavior can be observed, up to perfect training accuracy scores of up to 100%. The only exception is the genetic algorithm, as it underfits the data.

Furthermore, the genetic algorithm can identify the negative samples much better than any of the other algorithms. However, it fails to distinguish the positive samples well. This observation is expressed by the high precision and low recall scores. The reasons for this behavior are manifold and will be discussed later.

Most algorithms performed best, including 12 features, while only two performed better with only seven features. The relatively large number of features included indicates that many features influence the dependent variable. It also provides a possible explanation for why some algorithms encountered overfitting issues. Of the oversampling methods, SMOTE was six times favorable, SMOTE Borderline and -Tomek two times each. As there are almost twice as many negative samples as positive samples, the oversampling method, in this case, could have a significant impact on the performance. However, the results do not show indications of preference for any algorithm-oversampler combination yet.

While Table 1 shows the results of averages over 30 seeds, the Figure 15 shows the spread of the test data accuracy of the results presented. It gives further insight into the generalization ability of the algorithm in the shape of boxplot distributions. Optimally, there would be no difference in the results. However, as the data set is small, a different distribution of the samples within the training- and test sets of the data can be observed to make a difference to the algorithm's performance.

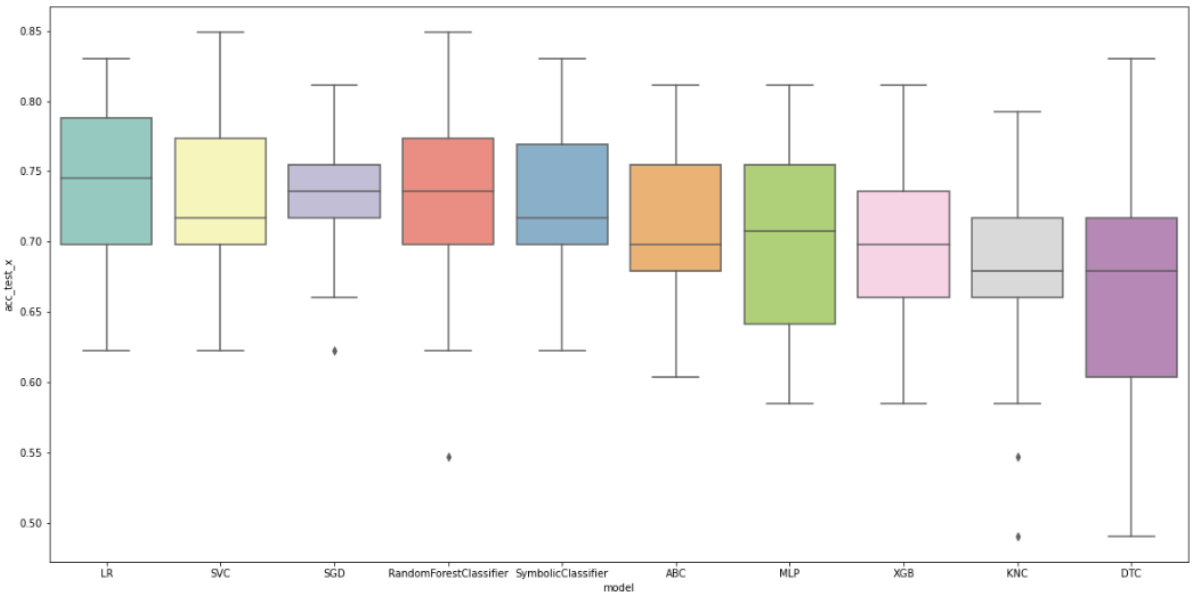

Figure 15 - result spread of the 30 test accuracy scores for each algorithm for the dependent variable "≤10"

The boxplots clearly show that there is frequently a spread of around 20 percentage points for almost all algorithms. Stochastic gradient descent is the exception, with a noticeably smaller spread compared to all other algorithms.

The statistical evaluation with the Wilcoxon signed-rank test, tested for a difference in the median of the results at a 5% confidence level. The test reveals, that there is no statistical median difference between the test accuracy scores of the logistic regression and either of the

support vector machine (W_{stat} = 120, p_{value} = 0.39), stochastic gradient descent(W_{stat} = 107, p_{value} = 0.34), random forest (W_{stat} = 145, p_{value} = 0.29) or genetic algorithm (W_{stat} = 156, p_{value} = 0.28). However, the logistic regression test accuracy scores and the adaptive boosting algorithm show a statistical significance (W_{stat} = 91, p_{value} = 0.018). Furthermore, the f1 scores of the logistic regression and the genetic algorithm show a statistically significant difference in median results with a W_{stat} = 27 and p_{value} = 0.000023.

The SHAP values displayed in Figure 16 are obtained using the tree explainer of the best performing logistic regression model. The SHAP values are in decreasing order of each features' importance. Each dot represents one sample in a color encoded for the value this samples value. The SHAP values show that the first five features influence the prediction. However, the age and primary tumor's influence are very limited or very specific to one sample and are not generalizable as the sample values have a SHAP value around zero . The SHAP values attribute breast tumor a very strong influence on the dependent variable. This can be interpreted as breast tumor having a high metastatic activity. If the lymph node is the first met organ site, the chance of developing more than ten lesions as a patient is lowered according to the SHAP values. Both these observations are in line with the medical understanding of the disease. The average burden and the cumulative burden increase the chance of developing more lesions with growing tumor size. This also holds for the SUV max of the first radiotherapy treatment.

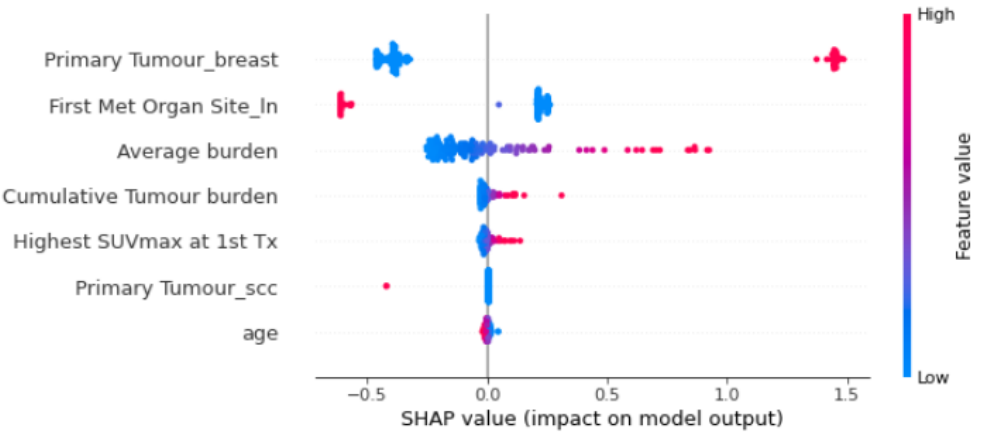

Figure 16 - SHAP values model explanation for the dependent variable "≤10"

An additional benefit of SHAP values can be observed in the age variable. As all values are centered on the null line, it can be discussed why this variable is considered. While there is a tendency that higher age of a patient has a negative impact on the development of the number of lesions, and vice versa, the impact seems marginal. Furthermore, it can be logically explained that a higher age could lead to the development of fewer than ten lesions for several reasons. First, there is less time left to develop ten lesions until life expires for a patient. Secondly, the immune system will be less capable of coping with ten lesions than the immune system of a younger patient.

A squamous cell carcinoma as the primary tumor has a negative impact on developing numerous lesions. However, one sample in the data is most likely not robust for this statement to be made. Hence, this variable also should be considered for exclusion or medical confirmation.

Dependent variable: “PMFS Oligo Status (≤5) maintained”:

The variable "PMFS Oligo Status (≤5) maintained" describes whether a patient remained within the oligometastatic status with less than five simultaneous lesions throughout his timespan in treatment. Once a patient crosses the border to the polymetastatic state (defined with six or more lesions simultaneously), treatment of this patient becomes more complicated. The method of SOMA cannot be further applied, as it is directed towards oligometastatic lesions only. Hence, this variable gives insight into the invasiveness, treatability, and curability of a patient.

Table 2 describes the performance and configurations of the best parameters of each algorithms' best configuration averaged over 30 seeds:

model	sampler	# of features	accuracy training	accuracy test	F1 score	precision	recall
GA	SMOTE Borderl.	4	0.71	0.7	0.69	0.67	0.72
RF	SMOTETomek	4	0.82	0.69	0.69	0.67	0.73
MLP	SMOTE Borderl.	7	0.73	0.69	0.69	0.66	0.72
LR	SMOTE Borderl.	7	0.72	0.69	0.69	0.65	0.74
ABC	SMOTE Borderl.	4	0.75	0.68	0.67	0.65	0.7
SVC	SMOTE Tomek	4	0.72	0.67	0.69	0.62	0.77
SGD	SMOTE Tomek	4	0.72	0.67	0.69	0.62	0.78
XGB	SMOTETomek	7	0.76	0.66	0.65	0.64	0.68
DTC	SMOTETomek	4	0.74	0.66	0.67	0.61	0.75
KNC	SMOTE Tomek	4	1	0.65	0.63	0.64	0.65

Table 2 - results table for the dependent variable “PMFS Oligo Status (≤5) maintained”

The mean accuracy scores on the test data are within a range of five percentage points, between 65% and 70%. This very balanced picture is extended over the precision, recall, and f1 scores, where no significant differentiating observations can be made concerning the algorithms performance. All algorithms have a slightly better recall than precision, which means that they better identify those samples that did not maintain their oligometastatic status. A slight overfitting behavior is indicated for all algorithms in the difference between test and training accuracy. Figure 17 shows a large spread in results further indicating limited robustness of the models. However, further analysis would be necessary to evaluate the extend of overfitting behavior.

Notable is that no algorithm worked best with the SMOTE oversampling method. Especially in this case, as the samples of the dependent variable are very balanced with 96 and 78 samples (of [0,1] respectively) in the entire dataset, the slight difference in the sample distribution should have only a minor influence the oversampling method. Most algorithms preferred the minimum amount of four features provided.

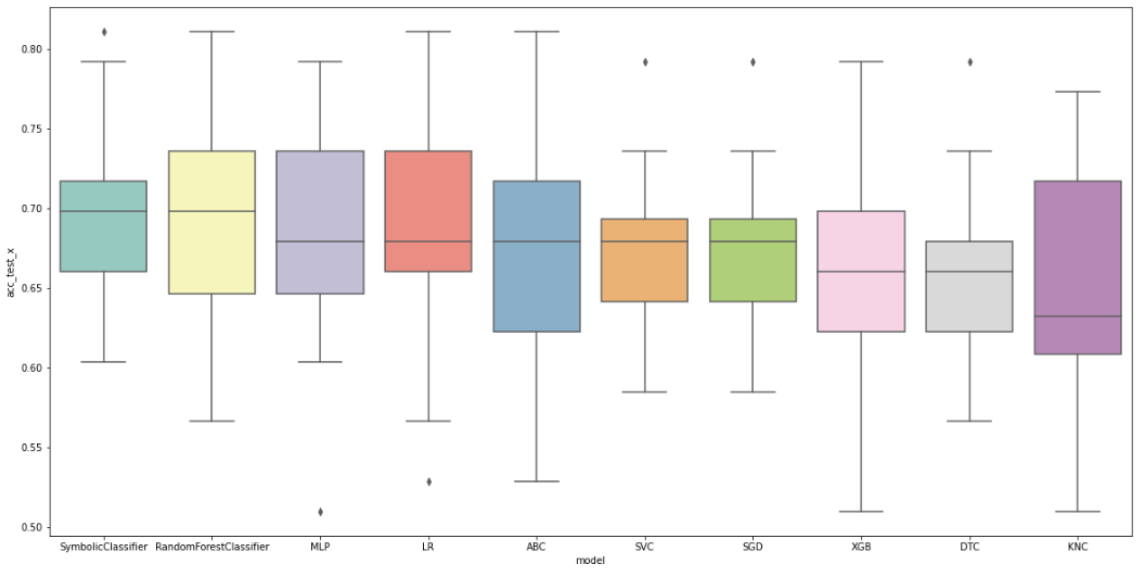

Figure 17 - result spread of the 30 test accuracy scores for each algorithm for the dependent variable " PMFS Oligo Status (≤5) maintained "

The spread of the algorithms test-accuracy scores varies heavily, with spreads from 15% points to almost 30% points and scores as low as 50% accuracy. In combination with the results presented previously in the table, it indicates a high sensitivity towards the data in the test set. This indicates that the generalization ability of the algorithms falls below the desired degree. The Support Vector Machine and Stochastic Gradient Descent have the lowest spread of results.

One possible explanation for this could be introduced noise in the data. This could be supported by the tendency of the high spread in results from the Adaptive Boosting and K-Nearest Neighbor classifiers, as both tend to be sensitive to noise in the data (Bootkrajang & Kabán, 2013).

The statistical evaluation with the Wilcoxon signed-rank test, tested for a difference in the median of the results at a 5% confidence level. The test reveals, that there is no statistical median difference between the test accuracy scores of the genetic algorithm and either of the random forest (W_{stat} = 96.5, p_{value} = 0.32), multi-layer perceptron(W_{stat} = 166, p_{value} = 0.39) or logistic regression (W_{stat} = 136, p_{value} = 0.20). However, the logistic regression test accuracy scores and the adaptive boosting algorithm show a statistical significance (W_{stat} = 83.5, p_{value} = 0.032).

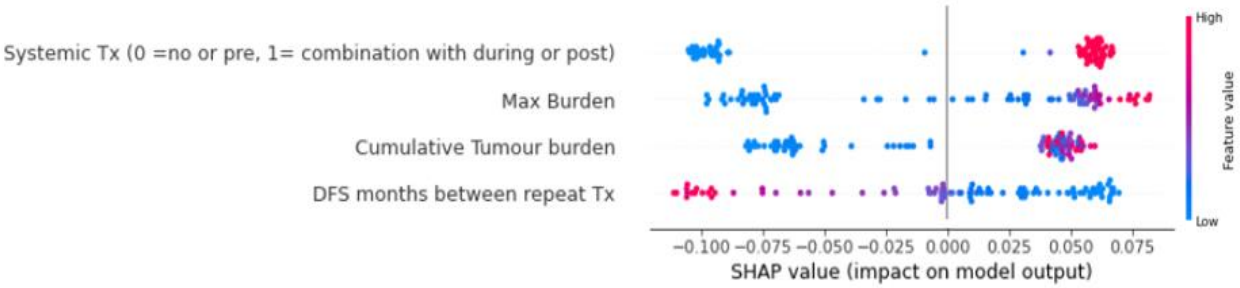

Figure 18 - SHAP values for the dependent variable " PMFS Oligo Status (≤5) maintained "

The SHAP values displayed in Figure 18 are obtained using the tree explainer of the best performing random forest model. The RF was selected more suitable for SHAP values as there is no statistical difference to the GA performance. Furthermore, the kernel explainers approximating nature can be mitigated.

The SHAP values indicate a significant influence of systemic therapy, a negative of a large maximum tumor burden and cumulative burden. This is very much in line with human intuition and current medical understanding. The disease-free months between treatments have an inverse impact on the dependent variable. However, it can be discussed whether this variable helps generate insights in this case. It does not help describe the progress of the treatment parameters but instead can be expected to correlate with the dependent variable. However, it seems that the algorithm is learning from the data sensibly. Further investigation of other variables influencing the dependent variable could generate additional insights into further interaction of variables.

Dependent variable: “OS Months”

The variable "OS months" explains the months of survival of a patient after the first examination as part of the program. Accurately forecasting the time of survival will increase the treatment options, maximizing the patient's life quality. In the context of SOMA-treated patients, especially the heterogeneity of the tumor and the small sample size opposes a significant challenge to accurate forecasts.

Table 3 describes the performance of the best parameters for each algorithms' best configurations mean over 30 seeds:

model	# of features	mean absolute error	mean squared error	root mean squared error	R^2
RF	4	14	291.72	17.07981	0.25
ABC	4	14.3	310.9	17.63236	0.2
GA	4	14.52	404.93	20.12287	-0.05
Lasso	7	15.68	339.72	18.43149	0.13
DTR	4	15.69	394.58	19.86404	-0.02
SVR	4	16.01	387.93	19.69594	0
KNR	7	16.08	402.37	20.05916	-0.04
XGB	4	35.56	1662.09	40.76874	-3.3
MLP	7	35.6	1664.94	40.80368	-3.31

Table 3 - results table for the dependent variable "OS months"

Among the nine algorithms examined, the mean absolute errors are between 14 and 35.6 months. However, seven of them are in the range between 14 and 16. This means that, on average, the prediction has an error of 14 months from the observed value. At an average survival of 36.4 months after the first treatment, this results in prediction being accurate to an average error of about 50%. The random forest algorithm generated the best results.

However, the root mean squared error might be the better error metric, as it increasingly penalizes larger errors. In the medical context, this translates to higher emphasis on a robust prediction. Compared to their mean absolute error, the support vector machine and the lasso regression have a relatively better root mean squared error. However, the random forest still performs best. This shows that they have fewer large errors, hence, a better generalization ability. It might be a result of the regularization ability of these two algorithms.

The R^2 value of the random forest is the best among the algorithms with a value of 0.25. This translates to the random forest's ability to explain the variance in the data set 25% better than the mean of the dependent variable. In other words, it only accounts for 25% of the variance in the data. This results in a minimal understanding of the data by the machine learning model. Five of the nine algorithms fail to have a positive R^2 value, which indicates that algorithms have difficulties learning from the data.

Figure 19 compares the root mean squared error of the models' best configurations results in over 30 seeds.

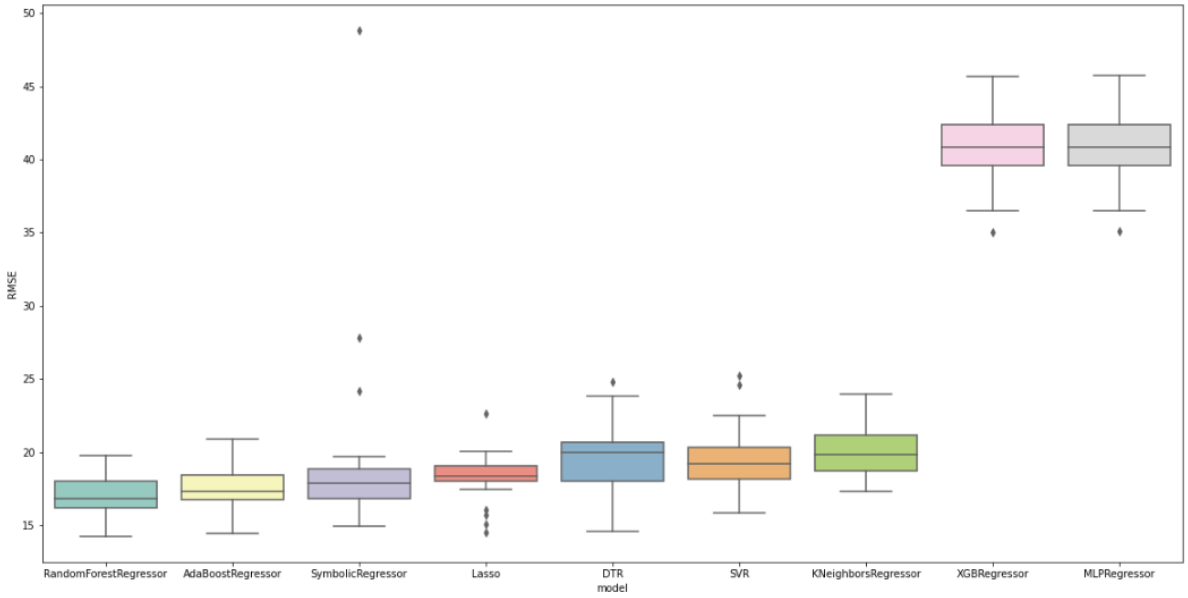

Figure 19 - result spread of the 30 test accuracy scores for each algorithm for the dependent variable " OS Months"

While the two best-performing algorithms, the random forest and adaptive boosting, both have an average distribution of results obtained, the genetic algorithm performs very well on most of the samples. However, it lacks generalization ability on a few. The opposite can be observed by the lasso regression, which has the smallest spread of prediction errors. However, a few outliers, predominantly on the positive side. This might be a result of the regularization term embedded in the algorithm, as mentioned before.

The statistical evaluation with the Wilcoxon signed-rank test, tested for a difference in the median of the results at a 5% confidence level. The test reveals a statistical median difference between the root mean squared error scores of the random forest and either of the adaptive boosting (W_{stat} = 45, p_{value} = 0.0001) and genetic algorithm (W_{stat} = 89, p_{value} = 0.003). The same holds for the difference in median R^2 scores of the random forest and the adaptive boosting algorithm (W_{stat} = 53, p_{value} = 0.002). Therefore, it can be concluded that the random forest algorithm statistically significantly outperforms the other algorithms in this specific setting.

The SHAP values displayed in Figure 20 below are obtained using the tree explainer of the best performing random forest model. Unsurprisingly the disease-free survival between repeated ration therapy sessions and the number of radiotherapy sessions for patience have the highest predictive power over the dependent variable. The samples of these two variables are fairly even distributed according to their feature value.

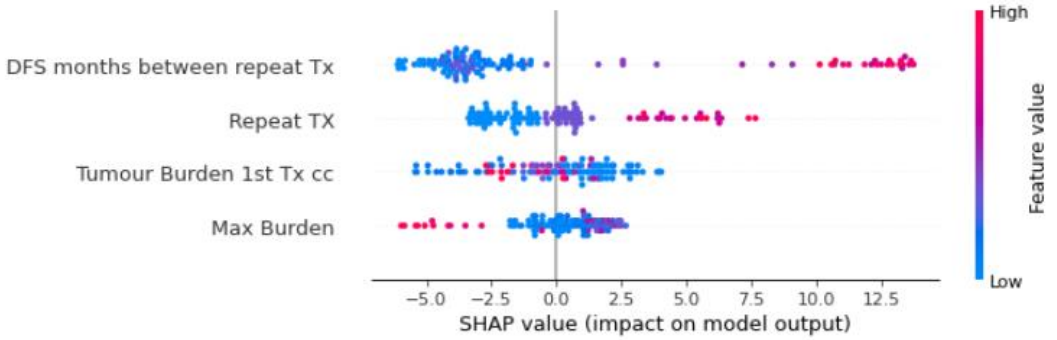

Figure 20 - SHAP values for the dependent variable " OS months"

The tumor burden at the first radiotherapy does not provide helpful information, as the distribution of the samples as SHAP values is very scattered. Although a large tumor burden could be expected to have a negative impact on the survival of a patient, the data does not provide this insight. Reasons for this could be interaction effects with other variables that are not further explained here. Furthermore, the SHAP values indicate that a sizeable maximum burden negatively influences a patient's survival. However, this does not necessarily always need to be the case.

In conclusion, the models feature importance, and insights generated with the SHAP values align with the medical intuition. This means that, despite the little predictive power of the model, it seems to understand some of the underlying patients' treatment mechanics. Further analysis might lead to detection of interaction effects. It should be discussed to exclude specific treatment process parameters for this analysis to derive more insights about the tumors heterogeneity impact on the dependent variable. This would exclude variables giving less insight about the disease despite relatedness.

Dependent variable: “PMFS Time to endpoint”

The variable "PMFS time to endpoint" explains the time in months a patient does not enter the polymetastatic status after the first examination. It addresses a similar need as the classification problem described previously, as polymetastatic cancer requires different treatment approaches, where SOMA is not applicable anymore. Knowing if, or when this timepoint could be reached potentially significantly impacts the treatment choices.

Table 4 describes the performance of the best parameters for each algorithms' best configurations mean over 30 seeds:

model	No features	Mean absolute error	root mean squared error	R^2
GP	7	10.8	15.79	0.33
RF	7	11.95	15.26	0.39
AdaBoost	12	12.75	16.23	0.3
DTR	12	12.89	17.69	0.18
SVR	7	12.94	17.99	0.16
Lasso	7	13.52	16.96	0.25
KNN	7	13.82	19.01	0.04
XGB	4	25.41	32.16	-1.73
MLP	7	25.45	32.19	-1.73

Table 4 - results table for the dependent variable " PMFS Time to endpoint "

The mean absolute errors of the algorithms are between 10.8 and 25.45, with an average value of the samples of 26.46. The XGBoost and the multi-layer perceptron network have difficulties learning from the data, as their performance is significantly worse than those of the other algorithms. The genetic algorithm delivers the best results for measuring the mean absolute error, followed by the random forest.

As already outlined previously, the root mean squared error is the better error measure in this context. Relative to this measure, the random forest algorithm performs best, with an error of 15.3 months. The random forest is also able to explain the most variance compared to the other algorithms. With an R^2 error of 0.39, the random forest can reduce the variance by 39% compared to the mean.

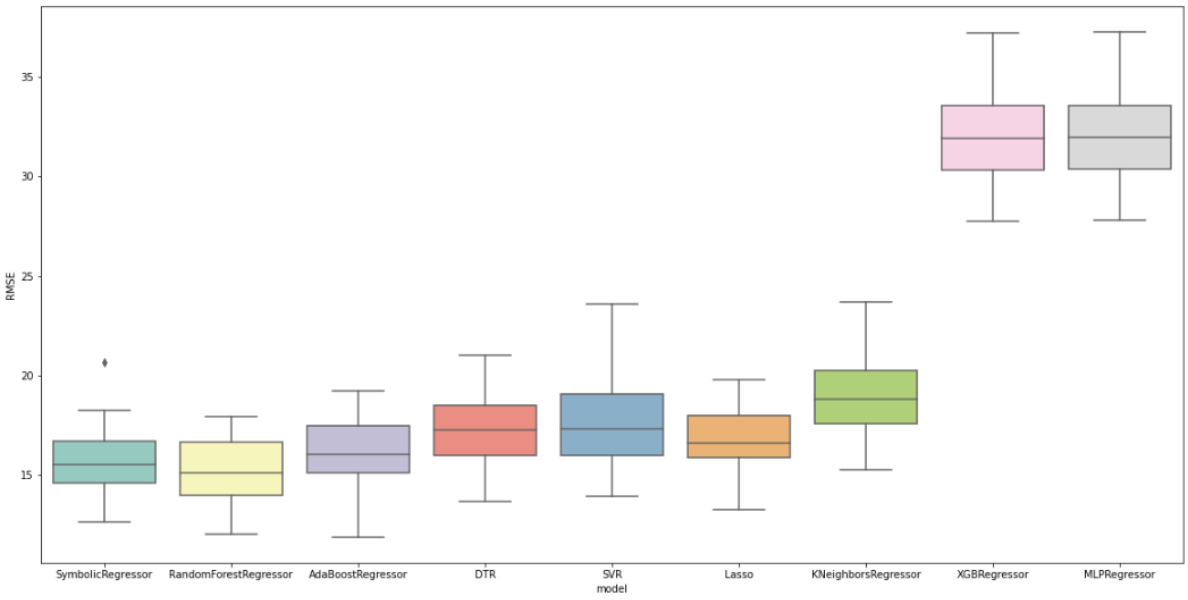

Figure 21 - result spread of the 30 test accuracy scores for each algorithm for the dependent variable "PMFS Time to endpoint"

Figure 21 shows the distribution of root mean squared errors from each algorithm's best configuration over 30 seeds. The genetic algorithm has one outlier and a slightly higher error distribution than the random forest, despite showing the smallest error distribution. Furthermore, the lasso regression proves its generalization ability with a low variance in errors. The support vector machine does not perform as well as the lasso regression despite the same regularization ability.

The statistical evaluation with the Wilcoxon signed-rank test, tested for a difference in the median of the results at a 5% confidence level. The test reveals no statistical median difference between the root mean squared error scores of the random forest and the genetic algorithm (W_{stat} = 203, p_{value} = 0.54). However, there is a statistical median difference between the random forest and adaptive boosting (W_{stat} = 59, p_{value} = 0003). The same holds for the difference in median R^2 scores (W_{stat} = 194, p_{value} = 0.42 and W_{stat} = 53, p_{value} = 0.0002 respectively). Therefore, it can be concluded that the random forest algorithm statistically significantly outperforms the other algorithms in this specific setting.

The SHAP values displayed in Figure 22 are obtained using the tree explainer of the best performing random forest model. The random forest model was chosen because of the best root mean squared error and R^2 score. The SHAP values indicate that the disease-free months between repeated radiotherapy have the highest predictive power. This follows the medical understanding of the disease as many disease-free months result from successful treatment or tumor inactivity.

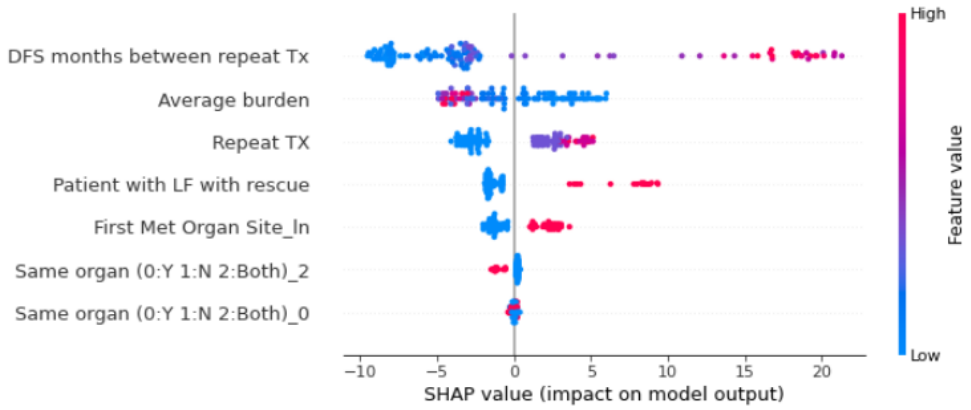

Figure 22 – SHAP values for the dependent variable " PMFS Time to endpoint"

The SHAP values also suggest that a high average burden leads to reduced polymetastasis free survival. A large number of radiotherapy sessions (Repeat TX) seem to prolong the polymetastasis free survival. In the case of the SOMA therapy method, this could indicate success in applying this method. The "Patient with LF with rescue" SHAP values express that patients who had to undergo rescue surgery have extended polymetastasis free survival. This observation could be biased as typically surgery is conducted when a high chance of cure is expected or as a last possible treatment measure.

In conclusion, the SHAP values follow the human intuition that the model seems to describe well. However, the lack of a high predictive power indicates that other variables need to be taken into account to forecast the dependent variable more accurately. The models cannot explain the data to a full extent.

Dependent variable: “Local Relapse”

The variable "Local Relapse Y=1 N=0" describes whether a lesion did relapse or not. For the prediction of this variable, only data until directly after the treatment of each lesion was used to increase the use of the prediction for the doctors. Knowledge about the relapse of a lesion directly after the radiotherapy treatment could provide great usefulness to the patient and the doctor in assuring the best treatment. One limitation of the data is the potential incompleteness of information. It cannot be known whether a lesion reoccurred after the most recent follow-up examination of one patient.

Furthermore the data is heavily imbalanced. Among the 605 lesions in the cleaned dataset, only 88 locally relapsed within the observation period. As good as this tremendous treatment success is for the patients, it poses a challenge from a machine learning perspective. The algorithms will tend to overfit in the case of multiple oversampled samples of each relapsed lesion, which is necessary to achieve a balanced training dataset.

Table 5 describes the performance of the best parameters for each algorithms' best configuration averaged over 30 seeds:

model	sampler	# of features	accuracy training	accuracy test	F1 score	precision	recall
SVC	SMOTE Borderl.	12	0.93	0.79	0.27	0.27	0.28
XGB	SMOTE	12	1	0.76	0.34	0.29	0.44
KNC	SMOTE Borderl.	12	1	0.75	0.34	0.27	0.46
RF	SMOTE	12	0.95	0.73	0.35	0.28	0.51
ABC	SMOTE	12	0.95	0.69	0.34	0.26	0.57
GA	SMOTE	12	0.63	0.67	0.22	0.2	0.34
SGD	SMOTE	12	0.51	0.6	0.15	0.15	0.39
LR	SMOTE	12	0.72	0.59	0.32	0.21	0.68
DTC	SMOTE Tomek	12	0.77	0.59	0.28	0.19	0.56
MLP	SMOTE Borderl.	7	0.52	0.32	0.23	0.17	0.77

Table 5 - results table for the dependent variable " Local Relapse "

The accuracy on the test dataset of all algorithms is in the range between 79% and 32%. For most algorithms, this result is obtained with an overfitting behavior of the model as the training set accuracy is better in most cases. In the circumstances of this particular dependent variable, special attention should be paid to recall, as it describes the ability of a model to detect the relapse of a lesion. Furthermore, as the data is highly imbalanced and relapsed lesions are scarce, the accuracy alone will skew the model's performance evaluation. The recall scores range from 0.28 to 0.77. The lowest recall score is obtained by the model with the best overall test accuracy. This concludes that the model well describes lesions that do not relapse but fails to identify lesions that relapse. The opposite can be stated from the multi-layer perceptron, which has the highest recall but lowest test accuracy, hence, forecasting most lesions to relapse. To balance the two scores, the f1 score should be considered to identify the best performing algorithm. For this measure, the random forest and adaptive boosting algorithm present the best performance.

The overfitting behavior could be expected because of the high imbalance of the dataset. This leads to the necessity to oversample the samples of the minority class multiple times to balance the training data set, leading to various similar samples. Very similar samples increase the difficulty of learning as algorithms tend to memorize the training data, as low variance leads to lower generalization ability of the model.

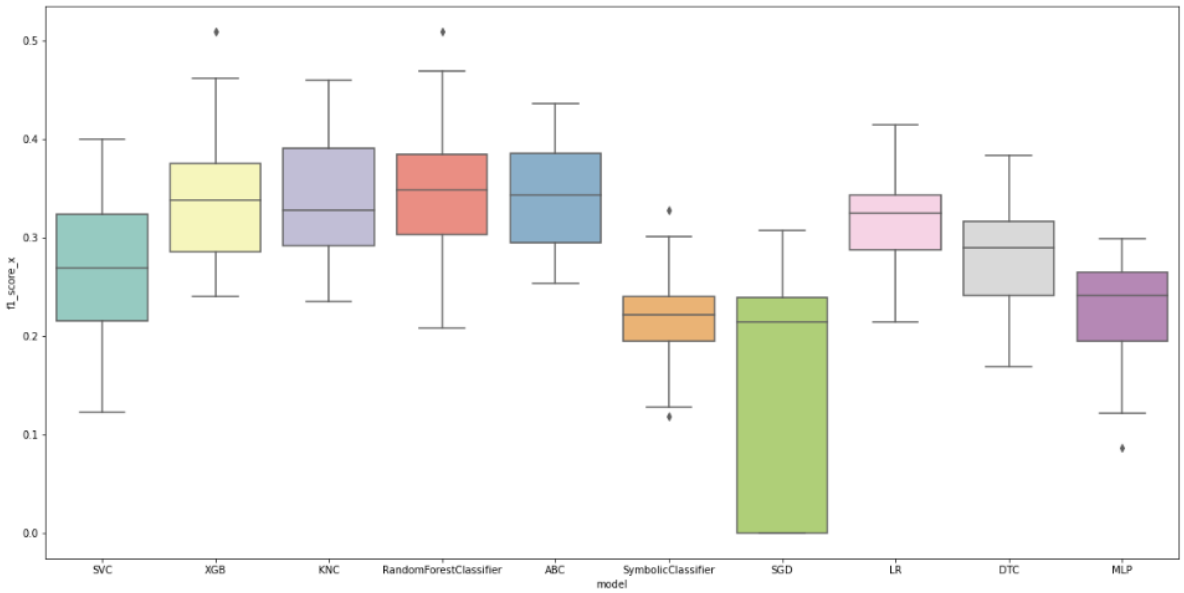

Figure 23 - result spread of the 30 test accuracy scores for each algorithm for the dependent variable "Local Relapse"

Figure 23 displays the spread of f1 scores over 30 seeds of the best performing configuration of each model. It can be observed that there is a wide variation in performance and variation within the scores between the different models. Because of the challenges this problem poses, a more tailored methodology to obtain more consistent results for each model might be required.

The adaptive boosting algorithm has the smallest spread of results and the best f1 score. The stochastic gradient descent algorithm has the most extensive spread of results within the configuration, indicating poor generalization ability in this scenario.

The statistical evaluation with the Wilcoxon signed-rank test, tested for a difference in the median of the results at a 5% confidence level. The test reveals, that there is no statistical median difference between the F1-scores of the random forest and either of the Adaptive Boosting (W_{stat} = 220, p_{value} = 0.79), K-Nearest Neighbor algorithm (W_{stat} = 203, p_{value} = 0.54) or XGBoost algorithm (W_{stat} = 196, p_{value} = 0.45. However, there is a statistical median difference between the Adaptive Boosting and Logistic Regression (W_{stat} = 95, p_{value} = 0.004).

The random forest model was selected to explain the models' performance, as it showed the best f1 score. The SHAP values explain that the random forest model gives the highest priority the mean grey (radiation) dose delivered to the lesion has the most substantial

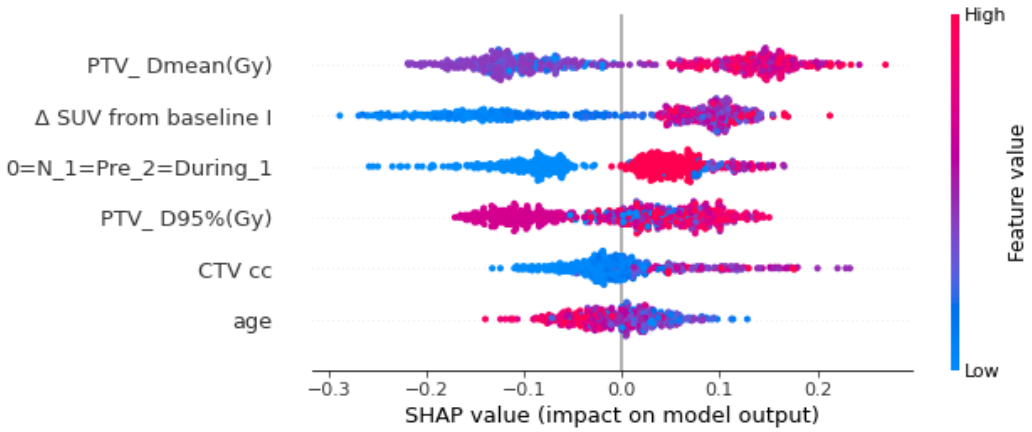

Figure 24 - SHAP values for the dependent variable "Local Relapse"

influence. Additionally, the mean dose delivered to 95% of the lesion (PTV_D95%) is considered. The deviation in SUV from the baseline is considered as the second most important feature. Surprisingly, not explicitly expressed in the research by Grecko et al. (Greco et al., 2019) on this variable, the systemic therapy before the treatment has undeniably impacted the model. Finally, the tumor size and age are taken into account by the random forest.

In this case, further analysis of the samples SHAP values in detail could reveal more insights, as the two radiation measurements and systemic therapy values could have interactivity, as the features are not linearly distributed by feature value. However, great alignment of the SHAP values with the human intuition and understanding of lesion relapse can be observed.

Dependent Variable: "LRFS Months"

The "LFRS Months" variable describes the local relapse-free months on a lesion level after radiation treatment of one specific lesion. It is an extension to the binary local relapse problem. Knowing, whether a lesion will relapse (reappear) is very helpful, as the lesion can be kept under close observation. Knowledge about the relapse of a lesion in the following months or years allows drawing further implications. Furthermore, it may be possible to pinpoint reasons for late or an early relapse of lesions.

The following table describes the performance of the best parameters for each algorithms' best configuration averaged over 30 seeds:

model	No features	Mean absolute error	Root mean squared error	R^2
RF	7	13	17.00	0.07
SVR	12	13.1	17.62	0
DTR	12	13.39	18.54	-0.11
GA	7	13.45	18.90	-0.16
Lasso	12	14.09	17.64	-0.01
KNR	12	14.59	19.37	-0.22
ABC	12	15.69	18.31	-0.09
MLP	7	22.11	28.27	-1.59
XGB	7	24.01	29.78	-1.87

Table 6 - results table for the dependent variable "LFRS months"

The mean absolute error of the obtained results for all algorithms ranges between 13 and 24 months. With an average of all samples of the dependent variable of 24.35 relapse-free months, these results have little predictive power. The best performing algorithms were the random forest, the support vector machines, and the decision tree algorithm. The root mean squared error should be the measure to assess the algorithm's performance for the same reasons as outlined before. Considering this, the lasso regression has a relatively better performance than in the comparison of the mean absolute error. However, it does not provide better results than the random forest. The multi-layer perceptron and the XGBoost algorithms have noticeably lower performance compared to the other algorithms.

Except for the random forest algorithm, the R^2 error for all algorithms is below or at zero. The random forest algorithm only explains 7% of the variance in the dataset compared to the mean. This concludes that the models poorly fit the data and do not explain the variance well.

All algorithms perform best with seven or 12 features. None of the algorithms achieved better results with only four features. This indicates that there is additional information to be obtained in the data. However, as the algorithms poorly explain the variance, the information required to forecast the dependent variable accurately might not be fully available. Much noise in the data might further increase the difficulty for the algorithms to learn. On the other hand, it might be the case that a higher degree of sophistication of the models is required to forecast the time until relapse accurately.

Figure 25 below shows the distribution of root mean squared errors. The observed errors are all within the range of about five error points for most algorithms. The adaptive boosting algorithm shows the lowest variance of errors within the range between 17 and 20. The decision tree, however, shows the most extensive spread in results. The genetic algorithm has one outlier, which could be an indication of a lack of robustness. On the other hand, the lasso regression once more improved relative to the evaluation comparison from the mean average error with a spread in results as small as most other algorithms.

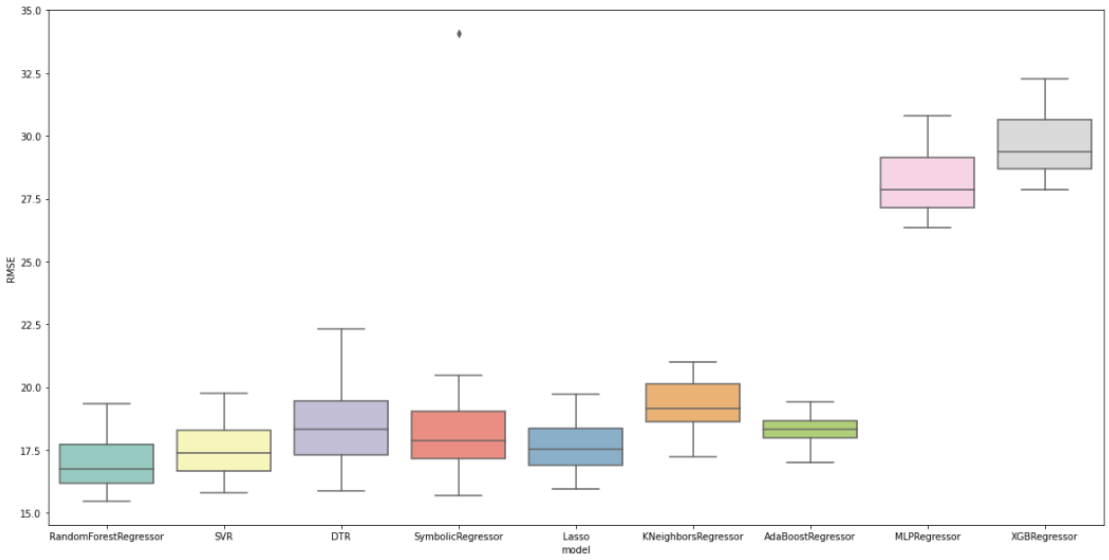

Figure 25 - result spread of the 30 test accuracy scores for each algorithm for the dependent variable "LFRS months"

The statistical evaluation with the Wilcoxon signed-rank test, tested for a difference in the median of the results at a 5% confidence level. The test reveals a statistical median difference between the root mean squared error scores of the random forest and either of the support vector machine (W_{stat} = 14, p_{value} = 0.00007) and lasso regression (W_{stat} = 9, p_{value} = 0.000004). The same holds for the difference in median R^2 scores of the random forest and the support vector machine (W_{stat} = 14, p_{value} = 0.002). Therefore, it can be concluded that the random forest algorithm statistically significantly outperforms the other algorithms in this specific setting.

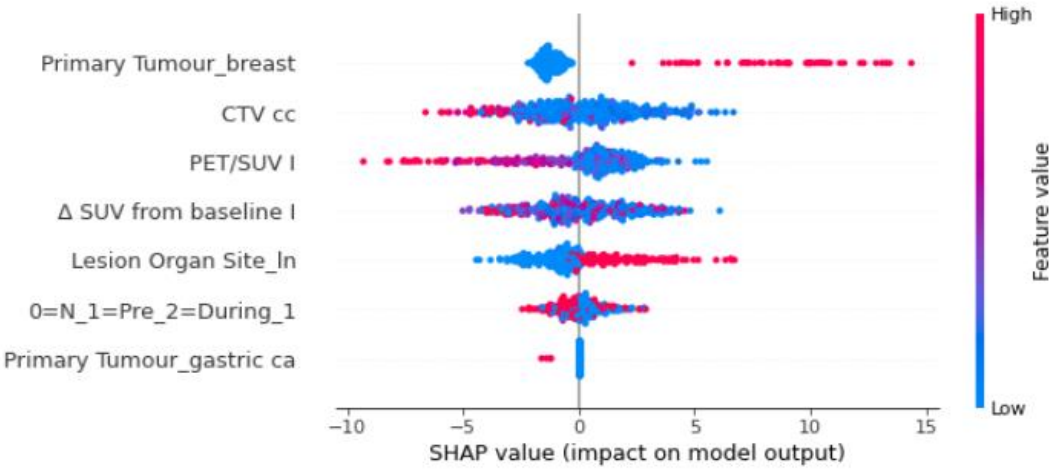

Figure 26 - SHAP values for the dependent variable "LFRS months"

The SHAP values displayed in Figure 26 are obtained using the tree explainer of the best performing random forest model. The SHAP values indicate that cancer in the breast as the primary tumor or in the lymph nodes as the lesion organ site does increase the time of relapse-free survival.

Contrary, a high CTV cc or PET/SUV I decrease the time for local relapse, indicating that larger or more active tumors are more prone to relapse. A few samples can be observed that do not follow this trend and could be suspect to noise or inability of the model to understand the data completely. Furthermore, the delta SUV from baseline I and the systemic therapy pre-radiation features have a very scattered distribution of samples. This makes it difficult to interpret these features. However, it is likely, that this can be a product of the poor predictive power of the model. On the other hand, interaction with other features could have led to this and could be explored further.

7. Discussion

The research goal of this thesis is to evaluate whether machine learning methods can deliver decision support for sequential single-dose radiation therapy to cure oligometastatic cancer lesions. Despite the constraints opposed by the data in terms of size, imbalance, and potential incompleteness, the results of this thesis show that machine learning algorithms can undoubtedly contribute to understanding patients' disease progress. In some scenarios, even a degree of predictive power could potentially be applied in practice, having the potential to evolve as a decision support system. However, in other scenarios, very poor predictive power does not offer the possibility to derive predictions from the algorithms. Likely, the decision support to unveiling more information about the disease progress will be limited in those cases. Most importantly, however, SHAP values seem to be a valuable addition in all scenarios researched in this thesis. Despite the poor performance, the algorithms identified relationships in the data aligned with human intuition.

On the one hand, SHAP values offer accurate insights into the decision-making process of a machine learning algorithm. They can potentially reveal interaction effects between variables that complex machine learning algorithms discover, which escape commonly applied univariate analysis in the medical field. This was already shown in a similar study on nonmetastatic nasopharyngeal carcinoma Du, Lee, et al. (Du et al., 2019). An in-depth analysis of the SHAP variables on a bi-variate level could make these relationships visible for SOMA. Furthermore, SHAP offers the possibility to include more complex machine learning models in practical medical applications. They bridge the gap between complex “black-box” machine learning models and the critical understanding of a machine learning model's functionality by doctors successfully.

Comparing the machine learning algorithms' performance over the six investigated problems shows that the random forest algorithm outperforms (or is insignificantly worse than the best alternative) the other algorithms in 5 out of 6 cases. This result is in line with previous research, as the random forest has the advantage of efficiently dealing with high dimensional data through incorporated feature selection, overfitting limitations through its low tree depth, and complex data structures by utilizing ensemble votes (Qi, 2012), (Y. Wu et al., 2020). However, other research found other algorithms, such as genetic programming or Support Vector Machines, favorable in similar problem settings (Vanneschi et al., 2011) (Ahmad et al., 2013). This concludes that the Random Forest provides a good starting point for analysis in the context of SOMA as it deals successfully with the problems outlined in the problem statement.

Surprisingly XGBoost did not perform well in comparison to other algorithms. Although it has shown good performance in other cases in similar problem settings on small cancer-related datasets (Koyasu, Nishio, Isoda, Nakamoto, & Togashi, 2020), (Nishio et al., 2018), it did not perform well on most problems presented. This is surprising, as the XGBoost algorithm is very similar to the random forest algorithm as it is an advanced implementation of gradient boosted decision trees. However, it is possible that no favorable hyper-parameter configuration was provided that the algorithm trains well on.

Furthermore, the multi-layer perceptron algorithm did not perform well on any of the provided problems. Although neural networks can analyze cause- and effect relationships especially well in complex systems, such as health, it is known that small datasets require a

different approach to train the model (Pasini, 2015b). Transfer learning, few-shot learning, or deep neural network architectures can provide approaches to mitigate this problem. However, they are not in the scope of this thesis research. This could be especially interesting in the context of SHAP values, which could display the potentially learned interaction effects of variables.

The comparison of the performance of the algorithm allows for one more conclusion to be made. For solving regression problems, algorithms with regularization ability had a smaller spread of results — the limited number of samples to learn from increase the impact of every individual sample. Hence, high dimensionality and potentially more noise from an increasing amount of features will make it increasingly difficult for an algorithm to develop robust predictions.

There are several limitations to the results presented in this thesis. These limitations apply to the underlying data, the methodology, and the interpretation of the results. One limitation that has already been mentioned is the potential incompleteness of data. Patients who do not have a complete data record in follow-up examinations will introduce noise to the data of unknown degrees. Among others, this is one possible explanation for the limited ability of the algorithms to obtain predictive power in the regression problems presented.

Furthermore, it limits the meaningfulness of specific results displayed. For example, the local relapse of the lesion was determined on data points collected up to 36 months after treatment, as the data does not allow for a larger timespan. Arguments can be made that the appropriate timespan to evaluate relapse of a lesion is two to ten years (AMLING et al., 2000). Furthermore, the treatment of outliers in the data could have led to different results. However, first explorations revealed that this would have unreasonably reduced the samples further.

The limitations of the methodology presented in this thesis are numerous. The performance comparison of the machine learning methods presented is not extensive. Numerous machine learning techniques can be applied to improve performance in data pre-processing, outlier removal, feature selection, and algorithm optimization. However, to have a reasonable setting for comparing the algorithms, the presented methodology provides several options for each algorithm to obtain data favorable for the algorithm. The results should provide criteria to evaluate how machine learning scenarios can add value to doctors' decisions. Furthermore, they can indicate techniques that are favorable for the underlying data.

In the future, the process of feature selection should be more aligned with the desired explanatory goal of the prediction. While some variables will correlate with others, it does not necessarily mean that they offer additional insights or benefit to the algorithm if included. Additionally, the feature selection method applied is a wrapper function, eliminating features not considered. Other techniques, such as Principle Component Analysis, reduce dimensionality while retaining variance in the data. In the context of small datasets trying to solve complex problems, this can be beneficial, as it enables to include all important information while reducing dimensionality as presented by Oikonomou (Oikonomou et al.).

One specific limitation of the interpretability of SHAP values must be clarified. SHAP values are only able to explain the patterns learned by the machine learning algorithm. They are not able to explain the underlying sample characteristics (Du et al., 2019). Hence, they

might deliver objectively wrong reasoning even in the case of perfect accuracy if the data allows for this. Additionally, in case of poor fit of the model, the SHAP values are only displaying the patterns learned by the machine learning model without making assumptions about the truthfulness of these patterns.

Nevertheless, human intuition can be a good indication. Especially as a decision support system in the medical field, this will not oppose real world a challenge. Eventually, it is not the data scientist taking medical decisions, but rather the doctor obtaining knowledge from the machine learning techniques applied.

The results of this thesis allow for two conclusions to be made. First, in most cases, machine learning methods can identify the relationship between observed treatment- and patient parameters and the desired dependent variable for SOMA radiotherapy treated patients or lesions. Furthermore, the SHAP values model explanations are generally in line with human intuition of the expected result and with the insights provided by univariate analysis of the features by Greco, et al. (Greco et al., 2019). However, the degree of accuracy and robustness of a model varies depending on the machine learning techniques applied.

Second, SHAP values offer great insights into understanding the mechanics of a machine learning model in medical applications. It has the potential to bridge the gap between theoretical "Black Box" models and the application of those in practice. Especially in the medical application where complex decisions have to be made taking various variables into account, SHAP values can extend machine learning methods. Rather than evaluating the decision support of a model from an accuracy-derived quality measure, it can explain how the decision was made for each patient. Furthermore, it could reveal complex interaction effects that more sophisticated machine learning models find but escape unimodal statistical analysis.

8. Future Research

Outside the scope of this thesis, the dependent variable of a lesion's local relapse was further explored and optimized with a different methodology. This dependent variable is already extensively analyzed by Greco et al. (Greco et al., 2019). The utilization of decision trees, pruned against assertive overfitting behavior and extensive feature selection, could improve the predictive power from the results displayed in this thesis. Additionally, an additional perspective with valuable insights about the features and their importance could be provided. This approach could be extended to the other dependent variables. The approach of this thesis methodology in obtaining a comparison of algorithms performance does not fully utilize the machine learning capabilities to the full extend, and many more insights are likely to be generated if applied.

The considerably lower predictive power of the XGBoost and the neural network can be further researched and certainly optimized. As the XGBoost is a development of the random forest algorithm, it should be possible to obtain the same results as the random forest model. Furthermore, deep neural networks have not been explored in this thesis but have offered promising results in similar contexts in other studies(Daoud & Mayo, 2019; N. Wu et al., 2019).

The application of SHAP values has not been utilized to the full possible extend. As Shapley values are calculated for every single sample, a bi-variate analysis could deliver valuable insights into the model's intuition of the data. The usefulness of this bivariate analysis was already proven in the study by Du, Lee et al. (Du et al., 2019).

9. Appendix

Description of variables in the Dataset:

Variable Name	Variable Description
Local Relapse Y(1) /N(0)	Did the treated metastasis reoccur in follow-up examinations? (binary)
LRFS Months	Relapse free time of single lesion in months
OS months	Time lived after the first treatment in months
PMFS Oligo Status (≤5) maintained until last FU 0=Y	Did the patient at any point in time develop more than five metastatic lesions?
PMFS Time to endpoint PM or no PM	Time in months before developing more than five metastatic lesions, death, or last examination
≤10	Did the patient develop more than ten metastatic lesions over the treatment timespan?
DoB	Birthdate
Gender	Gender
Primary Tumor	Location of primary Tumor
First Met Organ Site	Location of the first metastasis
CTV cc	Clinical target volume in cm^3
SUVmax Baseline PET-CT	Standardized Uptake Value quantifies the amount of tracer material uptake by the tumor tissue
Progression Elsewhere (Y:1/N:0)	Did a new metastasis occur after treatment?
Same organ (0:Y 1:N 2:Both)	Did a new metastasis occur after treatment affect the same organ, another, or both?
Systemic Tx (0 =no or pre, 1= combination with during or post)	Did the patient receive other treatment such as chemotherapy before, during, or after the first treatment?
First lesion(s) SDRT Only (1=Y)	Did the patient receive only Single-Dose Radiation Therapy for the treatment of the first lesion?
SOMA 1=Y 0=N	Did the patient receive sequential oligometastatic ablation therapy (SOMA)?
Number of targets at 1st Tx	Number of metastatic lesions at first treatment
Overall Regimen	Number of different therapy methods applied to the patient
Repeat TX	Number of overall radiation therapy session
Total number of targets	Total number of lesions in patients history since the first treatment
Total SOMA lesions	Total number of lesions treated with SOMA since the first treatment
N LRR	Number of locoregional recurrences

Patient with LF with rescue	Binary, if a lesion surgery of a patient was conducted
Tumor Burden 1st Tx cc	Sum of CTV cm^3 of all metastasis at first treatment
Tumor burden I SOMA	Sum of CTV cm^3 of all metastasis at first SOMA treatment
N of targets I SOMA	Number of metastasis at first SOMA treatment
Interval between ablations	Time in months between first and second treatment
Δ Tumor burden	Reduction in tumor mass in cm^3 between the first treatment and first SOMA treatment
Min Burden	Smallest tumor mass in cm^3 treated in one therapy session
Max Burden	Largest tumor mass in cm^3 treated in one therapy session
Average burden	Average tumor mass treated in one therapy session
Min %Δ Tumor burden	The smallest tumor mass delta in between two therapy sessions
Max %Δ Tumor burden	The largest tumor mass delta in between two therapy sessions
Mean %Δ Tumor burden	The average tumor mass delta in between two therapy sessions
Min N	The smallest number of lesions treated in one therapy session
Max N	The largest number of lesions treated in one therapy session
Min SOMA interval	The smallest interval in between two therapy sessions in months
Max SOMA Interval	The largest interval in between two therapy sessions in months
Average SOMA Interval	The average interval in between two therapy sessions in months
Cumulative Tumor burden	The total tumor mass in cm^3 for one patient over therapy timespan
Largest single SOMA burden	The largest tumor mass in cm^3 for one SOMA treatment
Largest single OM burden	The largest cumulative tumor mass in cm^3 at one point in time in oligometastatic-status
SOMA > 1st OM 0=N 1=Y	Was the tumor burden of a SOMA treatment larger than at the first treatment?
DFS months between repeat Tx	Disease free survival months until relapse or elsewhere progression of tumor
Highest SUVmax at 1st Tx	The highest Standarized Uptake Value of one metastasis at first treatment
Highest SUVmax ever	The highest Standarized Uptake Value of one metastasis over theramy timespan

N. of target organs at 1st Tx	The number of matastasis affected organs at first treatment
PTV_ Dmean	Average dose delivered to lesion in Gray
PTV_ D95%	Dosis delivered to 95% of the lesion in Grey

10. Bibliography

Ahmad, L. G., Eshlaghy, A., Poorebrahimi, A., Ebrahimi, M., & Razavi, A. (2013). Using three machine learning techniques for predicting breast cancer recurrence. *J Health Med Inform, 4*(124), 3.

AMLING, C. L., Blute, M. L., Bergstralh, E. J., Seay, T. M., Slezak, J., & Zincke, H. (2000). Long-term hazard of progression after radical prostatectomy for clinically localized prostate cancer: continued risk of biochemical failure after 5 years. *The Journal of urology, 164*(1), 101-105.

Batista, G. E. A. P. A. P., Ronaldo C., Monard, Maria Carolina2004). A study of the behavior of several methods for balancing machine learning training data. In (Vol. 6, pp. 20): Association for Computing Machinery (ACM).

Behravan, H., Hartikainen, J. M., Tengström, M., Kosma, V. M., & Mannermaa, A. (2020). Predicting breast cancer risk using interacting genetic and demographic factors and machine learning. *Scientific reports, 10*(1), 1-16.

Bi, Y., Xiang, D., Ge, Z., Li, F., Jia, C., & Song, J. (2020). An interpretable prediction model for identifying N7-methylguanosine sites based on XGBoost and SHAP. *Molecular Therapy-Nucleic Acids, 22*, 362-372.

Bootkrajang, J., & Kabán, A. (2013). Boosting in the presence of label noise. *arXiv preprint arXiv:1309.6818*.

Boser, B. E., Guyon, I. M., & Vapnik, V. N. (1992). *A training algorithm for optimal margin classifiers.* Paper presented at the Proceedings of the fifth annual workshop on Computational learning theory.

Breiman, L. (1996). Bagging predictors. In (Vol. 24, pp. 123-140): Springer Netherlands.

Breiman, L. (1997). *Arcing the edge*. Retrieved from

Breiman, L. (2001). Random forests. In (Vol. 45, pp. 5-32): Springer.

Breiman, L. e. a. (1998). *Classification and regression trees* (1. CRC Press repr. ed.). Boca Raton, Fla. u.a.: Chapman & Hall/CRC.

Bunkhumpornpat, C. S., Krung, Lursinsap, Chidchanok (2009). Safe-level-SMOTE: Safe-level-synthetic minority over-sampling technique for handling the class imbalanced problem. In (Vol. 5476 LNAI, pp. 475-482): Springer, Berlin, Heidelberg.

Chang, C.-C., & Lin, C.-J. (2011). LIBSVM: a library for support vector machines. *ACM transactions on intelligent systems and technology (TIST), 2*(3), 1-27.

Chawla, N. V. B., Kevin W, Hall, Lawrence O, Kegelmeyer, W Philip (2002). SMOTE: Synthetic Minority Over-sampling Technique. In (Vol. 16, pp. 321-357).

Chen, T., & Guestrin, C. (2016). *Xgboost: A scalable tree boosting system.* Paper presented at the Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining.

Cook, G. J. R., Siddique, M., Taylor, B. P., Yip, C., Chicklore, S., & Goh, V. (2014). Radiomics in PET: Principles and applications. In (Vol. 2, pp. 269-276): Springer-Verlag Italia s.r.l.

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning, 20*(3), 273-297. doi:10.1007/BF00994018

Cramer, J. S. (2002). The origins of logistic regression.

Curry, H. B. (1944). The method of steepest descent for non-linear minimization problems. *Quarterly of Applied Mathematics, 2*(3), 258-261.

Daoud, M., & Mayo, M. (2019). A survey of neural network-based cancer prediction models from microarray data. *Artificial Intelligence in Medicine, 97*, 204-214. doi:https://doi.org/10.1016/j.artmed.2019.01.006

Dayhoff, M. O., & National Biomedical Research, F. (1969). *Atlas of protein sequence and structure. [Vol. 1], [Vol. 1]*. Silver Spring [Md.]: National Biomedical Research Foundation.

de Hoon, M. J. L. I., S., Nolan, J., Miyano, S. (2004). Open source clustering software. In (Vol. 20, pp. 1453-1454): Bioinformatics.

Dietterich, T. B., Christopher, Heckerman, David, Jordan, Michael, Kearns, Michael, Introduction to Machine Learning Second Edition Adaptive Computation and Machine Learning. In.

Drucker, H., Burges, C. J., Kaufman, L., Smola, A., & Vapnik, V. (1997). Support vector regression machines. *Advances in neural information processing systems, 9*, 155-161.

Du, R., Lee, V. H., Yuan, H., Lam, K.-O., Pang, H. H., Chen, Y., . . . Vardhanabhuti, V. (2019). Radiomics Model to Predict Early Progression of Nonmetastatic Nasopharyngeal Carcinoma after Intensity Modulation Radiation Therapy: A Multicenter Study. *Radiology: Artificial Intelligence, 1*(4), e180075. doi:10.1148/ryai.2019180075

El Houby, M. F. E. (2018). A survey on applying machine learning techniques for management of diseases. *Journal of Applied Biomedicine, 16*(3), 165-174.

Fix, E., & Hodges, J. (1951). Discriminatory analysis, nonparametric discrimination.

Freund, Y., & Schapire, R. E. (1996). *Experiments with a new boosting algorithm.* Paper presented at the icml.

Freund, Y., & Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences, 55*(1), 119-139.

Friedman, J. H. (2001). Greedy function approximation: A gradient boostingmachine. *The Annals of Statistics, 29*(5), 1189-1232, 1144. Retrieved from https://doi.org/10.1214/aos/1013203451

Gillies, R. J., Kinahan, P. E., & Hricak, H. (2016). Radiomics: images are more than pictures, they are data. *Radiology, 278*(2), 563-577.

Goldberg, D. E. (1989). *Genetic algorithms in search, optimization, and machine learning*.

Greco, C., Pares, O., Pimentel, N., Louro, V., Morales, J., Nunes, B., . . . Fuks, Z. (2019). Phenotype-Oriented Ablation of Oligometastatic Cancer with Single Dose Radiation Therapy. *Int J Radiat Oncol Biol Phys, 104*(3), 593-603. doi:10.1016/j.ijrobp.2019.02.033

Grosu, A. L., Piert M Fau - Weber, W. A., Weber Wa Fau - Jeremic, B., Jeremic B Fau - Picchio, M., Picchio M Fau - Schratzenstaller, U., Schratzenstaller U Fau - Zimmermann, F. B., . . . Molls, M. Positron emission tomography for radiation treatment planning.

Han, H. W., Wen-Yuan, & Mao, B.-H. (2005). Borderline-SMOTE: A New Over-Sampling Method in Imbalanced Data Sets Learning.

Hebb, D. O. (1949). The organization of behavior; a neuropsycholocigal theory. *A Wiley Book in Clinical Psychology, 62*, 78.

Jansen, T., Geleijnse, G., Van Maaren, M., Hendriks, M. P., Ten Teije, A., & Moncada-Torres, A. Machine Learning Explainability in Breast Cancer Survival.

Karim, M. R., Cochez, M., Beyan, O., Decker, S., & Lange, C. (2019, 28-30 Oct. 2019). *OncoNetExplainer: Explainable Predictions of Cancer Types Based on Gene Expression Data.* Paper presented at the 2019 IEEE 19th International Conference on Bioinformatics and Bioengineering (BIBE).

Kégl, B. (2013). The return of AdaBoost. MH: multi-class Hamming trees. *arXiv preprint arXiv:1312.6086*.

Kohavi, R., & John, G. H. (1997). Wrappers for feature subset selection. *Artificial intelligence, 97*(1-2), 273-324.

Koyasu, S., Nishio, M., Isoda, H., Nakamoto, Y., & Togashi, K. (2020). Usefulness of gradient tree boosting for predicting histological subtype and EGFR mutation status of non-small cell lung cancer on 18F FDG-PET/CT. *Annals of Nuclear Medicine, 34*(1), 49-57. doi:10.1007/s12149-019-01414-0

Kumar, V., Gu, Y., Basu, S., Berglund, A., Eschrich, S. A., Schabath, M. B., . . . Gillies, R. J. (2012). Radiomics: the process and the challenges. *Magnetic resonance imaging, 30*(9), 1234-1248. doi:10.1016/j.mri.2012.06.010

Lambin, P., Rios-Velazquez E Fau - Leijenaar, R., Leijenaar R Fau - Carvalho, S., Carvalho S Fau - van Stiphout, R. G. P. M., van Stiphout Rg Fau - Granton, P., Granton P Fau - Zegers, C. M. L., . . . Aerts, H. J. Radiomics: extracting more information from medical images using advanced feature analysis. (1879-0852 (Electronic)).

Lambin, P., Rios-Velazquez, E., Leijenaar, R., Carvalho, S., van Stiphout, R. G. P. M., Granton, P., . . . Aerts, H. J. W. L. (2012). Radiomics: extracting more information from medical images using advanced feature analysis. *European journal of cancer (Oxford, England : 1990), 48*(4), 441-446. doi:10.1016/j.ejca.2011.11.036

Lemaréchal, C. (2012). Cauchy and the gradient method. *Doc Math Extra, 251*(254), 10.

Liao, S.-C., & Lee, I.-N. (2002). Appropriate medical data categorization for data mining classification techniques. *Medical informatics and the Internet in medicine, 27*(1), 59-67.

Liu, Z., Wang, S., Dong, D., Wei, J., Fang, C., Zhou, X., . . . Tian, J. The Applications of Radiomics in Precision Diagnosis and Treatment of Oncology: Opportunities and Challenges.

Lovell, M. (1983). Data Mining. *The Review of Economics and Statistics, 65*(1), 1-12. Retrieved from https://EconPapers.repec.org/RePEc:tpr:restat:v:65:y:1983:i:1:p:1-12

Lundberg, S., & Lee, S.-I. (2017). A unified approach to interpreting model predictions.

McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*

Mitchell, T., & McGraw-Hill. (1997). Machine Learning. In: New York: McGraw-Hill, Inc.

Mohri, M., Rostamizadeh, A., & Talwalkar, A. (2018). *Foundations of machine learning*: MIT press.

Mokhtari, K. E., Higdon, B. P., & Başar, A. (2019). *Interpreting financial time series with SHAP values.* Paper presented at the Proceedings of the 29th Annual International Conference on Computer Science and Software Engineering.

Nathalie Japkowicz, M. S. Evaluating Learning Algorithms: A Classification Perspective - - Google Books. In.

Nishio, M., Nishizawa, M., Sugiyama, O., Kojima, R., Yakami, M., Kuroda, T., & Togashi, K. (2018). Computer-aided diagnosis of lung nodule using gradient tree boosting and Bayesian optimization. *PloS one, 13*(4), e0195875.

Oikonomou, A. A.-O. X., Khalvati, F., Tyrrell, P. N., Haider, M. A., Tarique, U., Jimenez-Juan, L., . . . Cheung, P. Radiomics analysis at PET/CT contributes to prognosis of recurrence and survival in lung cancer treated with stereotactic body radiotherapy.

Parmar, C., Grossmann, P., Bussink, J., Lambin, P., & Aerts, H. J. (2015). Machine learning methods for quantitative radiomic biomarkers. *Scientific reports, 5*(1), 1-11.

Parsa, A. B., Movahedi, A., Taghipour, H., Derrible, S., & Mohammadian, A. (2020). Toward safer highways, application of XGBoost and SHAP for real-time accident detection and feature analysis. *Accident Analysis & Prevention, 136*, 105405. doi:https://doi.org/10.1016/j.aap.2019.105405

Pasini, A. (2015a). Artificial neural networks for small dataset analysis. *Journal of thoracic disease, 7*(5), 953.

Pasini, A. (2015b). Artificial neural networks for small dataset analysis. *Journal of thoracic disease, 7*(5), 953-960. doi:10.3978/j.issn.2072-1439.2015.04.61

Pasupa, K., & Sunhem, W. (2016). *A comparison between shallow and deep architecture classifiers on small dataset.* Paper presented at the 2016 8th International Conference on Information Technology and Electrical Engineering (ICITEE).

Prati, R. C. B., Gustavo E.A.P.A., Monard, Maria Carolina,(2004). Learning with class skews and small disjuncts. In (Vol. 3171, pp. 296-306): Springer, Berlin, Heidelberg.

Qi, Y. (2012). Random forest for bioinformatics. In *Ensemble machine learning* (pp. 307-323): Springer.

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). Model-agnostic interpretability of machine learning. *arXiv preprint arXiv:1606.05386*.

Robbins, H., & Monro, S. (1951). A stochastic approximation method. *The annals of mathematical statistics*, 400-407.

Rosenblatt, F. (1962). Principles of Neurodynamics: Perceptrons and the Theory of. *Brain Mechanisms*, 555-559.

Samuel, A. L. (1959). Some studies in machine learning using the game of checkers. *IBM Journal of research and development, 3*(3), 210-229.

Santosa, F., & Symes, W. W. (1986). Linear Inversion of Band-Limited Reflection Seismograms. *SIAM Journal on Scientific and Statistical Computing, 7*(4)

Shapley, L. S. (1953). A value for n-person games. *Contributions to the Theory of Games, 2*(28), 307-317.

Shrikumar, A., Greenside, P., & Kundaje, A. (2017). *Learning important features through propagating activation differences.* Paper presented at the International Conference on Machine Learning.

Shrikumar, A., Greenside, P., Shcherbina, A., & Kundaje, A. (2016). Not just a black box: Learning important features through propagating activation differences. *arXiv preprint arXiv:1605.01713*.

Tan, A. C., & Gilbert, D. (2003). An empirical comparison of supervised machine learning techniques in bioinformatics.

Toh, C., & Brody, J. P. (2021). Genetic Risk Score for Predicting Schizophrenia Using Human Chromosomal-Scale Length Variation.

Tomek, I. (1976). TWO MODIFICATIONS OF CNN. In (Vol. SMC-6, pp. 769-772).

Uddin, S., Khan, A., Hossain, M. E., & Moni, M. A. (2019). Comparing different supervised machine learning algorithms for disease prediction. *BMC Medical Informatics and Decision Making, 19*(1), 281. doi:10.1186/s12911-019-1004-8

Vallières, M., Freeman Cr Fau - Skamene, S. R., Skamene Sr Fau - El Naqa, I., & El Naqa, I. A radiomics model from joint FDG-PET and MRI texture features for the prediction of lung metastases in soft-tissue sarcomas of the extremities.

Vanneschi, L., Farinaccio, A., Mauri, G., Antoniotti, M., Provero, P., & Giacobini, M. (2011). A comparison of machine learning techniques for survival prediction in breast cancer. *BioData mining, 4*(1), 1-13.

Werbos, P. J. (1975). *Beyond regression : new tools for prediction and analysis in the behavioral sciences.*

Wilcoxon, F. (1945). Individual Comparisons by Ranking Methods. *Biometrics Bulletin, 1*(6), 80-83.

Wolpert, D., & Macready, W. (1996). No Free Lunch Theorems for Search.

Wu, J., Tha, K. K., Xing, L., & Li, R. Radiomics and radiogenomics for precision radiotherapy. (1349-9157 (Electronic)).

Wu, N., Phang, J., Park, J., Shen, Y., Huang, Z., Zorin, M., . . . Kim, E. (2019). Deep neural networks improve radiologists' performance in breast cancer screening. *IEEE transactions on medical imaging, 39*(4), 1184-1194.

Wu, Y., Liu, J., Han, C., Liu, X., Chong, Y., Wang, Z., . . . Li, S. (2020). Preoperative Prediction of Lymph Node Metastasis in Patients With Early-T-Stage Non-small Cell Lung Cancer by Machine Learning Algorithms. *Frontiers in Oncology, 10*(743).

Zelefsky, M. J., Yamada, Y., Greco, C., Lis, E., Schöder, H., Lobaugh, S., . . . Fuks, Z. Phase 3 Multi-Center, Prospective, Randomized Trial Comparing Single-Dose 24 Gy Radiation Therapy to a 3-Fraction SBRT Regimen in the Treatment of Oligometastatic Cancer.