Predicting acute radiation-induced toxicity in lung cancer patients undergoing radiotherapy on the basis of pre-treatment CT scans and dose distributions using machine learning

Msc thesis - Artificial Intelligence

Author:
Sjors Witteveen
(Solid-id: 4174518)
s.witteveen@students.uu.nl

Utrecht University
Department of Information and Computing Sciences
Supervisors:
Dr. Mel Chekol
Prof. Dr. Yannis Velegrakis

University Medical Center Utrecht
Department of Radiation Oncology[1]
Department of Radiology[2]
Supervisors:
Dr. Joost Verhoeff[1]
Dr. Jacquelien Pomp[1]
Prof. Dr. Ir. Nico van den Berg[1]
Wouter van Amsterdam, M.D., PhD (c)[2]

April 19, 2021

Abstract

Purpose: In this study, we used machine learning to predict acute radiation-induced toxicity for lung cancer patients undergoing radiotherapy on the basis of pre-treatment CT scans, contouring of organs, and radiotherapy dose distributions. The radiation oncologists can then use the acute toxicity prediction to create a treatment plan best suited for a patient.

Methodology: We prepared a data set that is larger than any other data set found in related work predicting radiation-induced toxicity in lung cancer patients. This data set consists of clinical features, dosimetrics, radiomics, and acute toxicity labels of 458 patients. The preprocessing of the data set is described in detail, which is essential for reproduction in future studies. Three classification algorithms were used in this study: Logistic Regression (LR), Random Forest (RF), and Support Vector Machine (SVM). Ridge regularization was used for both LR and SVM. The outcome was divided into four classes: *any toxicity*, *fatigue*, *esophagus toxicity*, and *lung toxicity.* For each outcome class, only dosimetrics and radiomics of organs related to the type of acute toxicity were used. A model was trained and evaluated for every combination of input feature type, classification algorithm, and output class. A nested 5-fold cross-validation was used to train the models and a grid search was performed to train the hyperparameters of each classification algorithm.

Results/conclusion: The best AUC scores for the outcome classes *any toxicity*, *fatigue*, *esophagus toxicity*, and *lung toxicity* were 0.72, 0.65, 0.86, and 0.57, respectively. Although difficult to directly compare AUC values due to the usage of different data sets, the AUC score of 0.86 is so far unmatched by related studies. For all classes except *lung toxicity*, the best AUC score was achieved using a LR model. The *lung toxicity* class was best predicted using RF, but this was likely due a deficiency in the input features requiring a nonlinear solution, rather than due to the nature of the class itself. Dosimetrics were found to be the most predictive features among the input types used in this study, followed by clinical features which performed slightly worse overall. The shape and first order radiomics were found to have almost no effect on the prediction of acute toxicity. Overall, the methods presented in this study are promising for the future of predicting acute toxicity.

Contents

1 Introduction

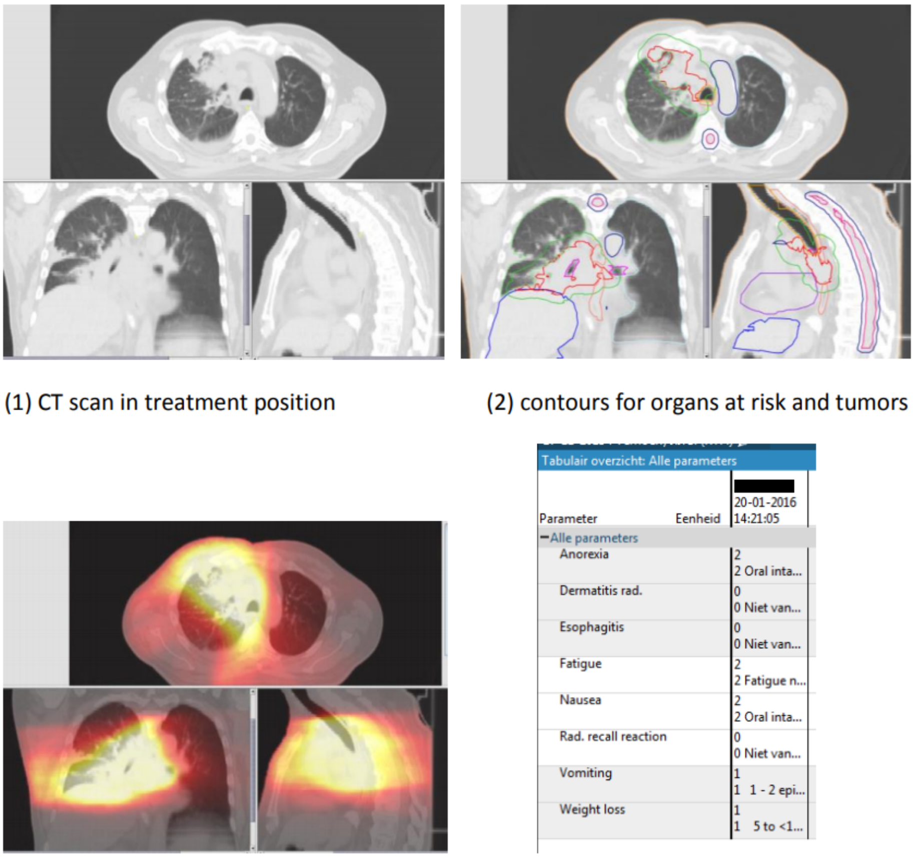
Lung cancer is a devastating disease with a dismal prognosis. The term lung cancer refers to all types of cancer originating from lung tissue. It is characterized by uncontrolled cell growth in tissues of the lung, resulting in the development of a tumor. Malignant lung tumors can metastasize outside the thorax, reaching a state where local treatment cannot contribute to cure. There are four different stages of cancer. In stage I, the cancer is localized in one part of the lung, making it accessible for surgery. In stages II and III, the cancer is locally advanced, with involvement of hilar and mediastinal lymph nodes. In stage IV, the cancer has spread outside the thorax and local treatment is not curative.

Over 13.000 patients are diagnosed with lung cancer every year in The Netherlands, most of which have stage III or IV. Patients with stage III are offered a combination of chemotherapy and radiotherapy if their condition is suitable for this heavy treatment. These treatment modalities with curative intent come with side effects. Chemotherapy has a systemic toxic profile. Radiotherapy, when given in high dose, can cause serious collateral damage. Therefore, the prescribed total dose is limited by the tolerance of the normal tissues surrounding the tumour and lymph nodes. In case of lung cancer, the main limiting organs at risk are the healthy lung, the esophagus, and the heart. Depending on the grade of toxicity, they can have a serious impact on the quality of life (QOL) of a patient and can even result in death.

Two forms of toxicities that can be caused by radiotherapy of the lung are pneumonitis and esophagitis. Radiation pneumonitis (RP) is an inflammation of lung tissue, which can result in a dry cough and shortness of breath. Esophagitis is an inflammatory reaction of the esophagus, that often comes with complaints of swallowing pain and obstruction. The heart toxicities are not well defined. This is because there is little known about acute heart toxicity and patients often have interfering cardiovascular disease, making it difficult to discriminate between radiation-induced toxicities and toxicities explained by pre-treatment comorbidities.

The Department of Radiotherapy at the University Medical Center Utrecht (UMCU) treats yearly around 250 stage I-II, 100 stage III, and 100 stage IV lung cancer patients, referred by pulmonologists from UMCU, Antonius, Diakonessenhuis, Meander and other hospitals. For patients treated with high dose radiotherapy, we used patient reported outcome measurement (PROM) to evaluate the toxicity. Toxicity is defined as toxicity negatively affecting the QOL of the patient. Whether a toxicity is considered to have a negative impact on the QOL of the patient depends on the grade of toxicity. The grading system is elaborated on in section 2.1. The subject of matter for this project is the acute toxicity in particular. Acute toxicity is toxicity that occurs within the acute period of one week after treatment start till three months after treatment start. The first week of toxicity registrations after the start of the treatment is not considered, as toxicities registered within this period often include toxicities that are already present before the treatment.

To improve the quality of treatment of lung cancer patients, we would like to have an idea of the acute toxicity that will be caused by radiotherapy of the individual patient before they are treated. To realize this, we would like to make a prediction of the acute radiation-induced toxicity using only patient data that is available pre-treatment. From around 3000 patients treated for lung cancer from 2010 to 2018, we have archives with (1) the planning CT scan, (2) contouring of organs at risk and tumors, (3) radiotherapy dose distribution and (4) toxicity registration (PROMs) (Figure 1). The goal of this project is to create a prediction model by applying methods from machine learning to these data. The outcome of our prediction model will be whether toxicity will occur or not rather than a specific grade of toxicity. This decision for a binary classification was made mainly because the toxicities are patient-reported. Therefore, the grading of toxicities in our data set is rather subjective. In addition, in investigating whether predicting acute toxicity is possible, predicting a



(1) CT scan in treatment position

(2) contours for organs at risk and tumors

Tabulair overzicht: Alle parameters

Parameter	Eenheid	[redacted] 20-01-2016 14:21:05
−Alle parameters		
Anorexia		2 2 Oral inta...
Dermatitis rad.		0 0 Niet van...
Esophagitis		0 0 Niet van...
Fatigue		2 2 Fatigue n...
Nausea		2 2 Oral inta...
Rad. recall reaction		0 0 Niet van...
Vomiting		1 1 1 - 2 epi...
Weight loss		1 1 5 to <1...

(3) Radiotherapy 3D dose distributions.

(4) toxicity registration

Figure 1: The four types of data available.

binary outcome is a sensible first step. Predicting a specific grade is a more difficult task that could be interesting for future work. The research question is the following:

How accurately can we predict acute radiation-induced toxicity for lung cancer patients undergoing radiotherapy on the basis of pre-treatment CT scans, contouring of organs, and radiotherapy dose distributions using machine learning?

To answer this question, we have used several types of features as input that have proven themselves to be predictive in previous studies that will be discussed in section 3. These types of features are clinical features, dosimetrics, and radiomics. Medical data are generally very unorganized, hence the preprocessing was a very substantial part of this project. These features are used as input for three classification algorithms: Logistic Regression (LR), Random Forest (RF) and Support Vector Machine (SVM). LR and SVM make use of Ridge (L2) regularization. All models are trained using a nested 5-fold cross-validation, performing a grid search to train the model's hyperparameters. The acute toxicity outcome that is to be predicted by the models is divided into four classes: *any toxicity*, *fatigue*, *esophagus toxicity*, and *lung toxicity*. Each outcome is predicted separately and for each class, only input that is relevant for the prediction of the particular type of toxicity is used.

Every possible combination of the different types of input and output is used to train every type of classification algorithm. This allows us to better compare the effect of each type of input separately or combined, the accuracy of which we can predict each class, and how well each classification algorithm performs in combination with different types of input and output.

The grading of toxicities, the input features, the classification algorithms, and the evaluation strategies are explained in section 2. The advances that have been made in related work regarding the prediction of toxicity are discussed in section 3. The data and their preprocessing, along with the detailed implementation of the classification algorithms are elaborated upon in section 4. The results are presented in section 5. In section 6, the results will be compared to related work, strengths and weaknesses of this study will be discussed, and suggestions for future work will be made. Final conclusions will be drawn in section 7.

2 State Of The Art

There is a large variety of classification algorithms and different types of input features that are predictive of acute toxicity. Below, the acute toxicity outcome is further defined by introducing a toxicity grading system (section 2.1). For the purpose of predicting acute radiation-induced toxicity, we will discuss three types of input features: clinical features, dosimetrics, and radiomics (section 2.2). The classification algorithms that have been selected for this project are LR, RF, and SVM (section 2.3), because their performance has been widely demonstrated (Bishop, 2006). Finally, the evaluation strategies will be discussed (section 2.4).

2.1 Toxicity Grading System

The grading of toxicities is done according to the Common Terminology Criteria of Adverse Effects (CTCAE), formerly known as Common Toxicity Criteria (CTC) [1]. In the CTCAE, a grading scale is provided for each Adverse Event (AE). In the latest version (v5.0) of this grading system, the grades are from 0 to 5, where in general, the grades mean the following:

Grade 0: None.

Grade 1: Mild; asymptomatic or mild symptoms; clinical or diagnostic observations only; intervention not indicated.

Grade 2: Moderate; minimal, local or noninvasive intervention indicated; limiting age-appropriate instrumental activities of daily living.

Grade 3: Severe or medically significant but not immediately life-threatening; hospitalization or prolongation of hospitalization indicated; disabling; limiting self care activities of daily living.

Grade 4: Life-threatening consequences; urgent intervention indicated.

Grade 5: Death related to Adverse Event.

Grade 1 toxicities do not affect the patient's QOL significantly. Toxicities of at least grade 2 are more harmful and can often negatively affect the patient's QOL. Hence, we consider patients to have experienced acute toxicity only if the grade of toxicity is at least 2.

2.2 Input Features

As stated in the introduction, the input features are taken from the patient's CT scan, contouring of organs, and radiotherapy dose distribution. Dosimetrics and radiomics are taken from the dose distribution and CT scan, respectively, where the contouring of organs is used to determine what part of the image is used to calculate the metrics for a specific organ. In addition, more basic information about the patient and treatment can be used for the prediction of acute toxicity, which we call the clinical features.

[1] `https://ctep.cancer.gov/protocolDevelopment/electronic_applications/ctc.htm`

2.2.1 Clinical Features

The clinical features are very simple data about the patient and their treatment. Three widely used clinical features are the age and sex of the patient and their stage of cancer. One can imagine, for instance, that in general, older people are more susceptible to toxicity than younger people. Other clinical features can be more treatment-related, such as the total radiation dose or the number of fractions the dose is divided in. A higher dose can result in more toxicity, but when it is divided over many fractions, generally less toxicity will occur.

A measure called the Equivalent Dose in 2 Gray (Gy) fractions (EQD_2) can be calculated for this (Fowler, 1989). This is a measure for the biologically equivalent dose delivered in fractions of 2 Gy and can be calculated using the following formula, where D stands for the total dose in Gy, d for the dose in Gy delivered per fraction:

$$\text{EQD}_2 = D \cdot \frac{d + \alpha/\beta}{2 + \alpha/\beta} \tag{1}$$

Evidently, the dose delivered per fraction (d) is calculated by dividing the total dose (D) by the total number of fractions. The α/β ratio is a measure of fractionation sensitivity. The lower the α/β ratio, the more susceptible to radiation damage a tissue is when increasing fraction dose. It is found in practice that if we deliver a total dose of 60 Gy in only ten fractions, the biological impact is much larger than if we deliver the same total dose of 60 Gy in 30 fractions. The EQD_2 of the former is equal to 108 Gy, while the EQD_2 of the latter is equal to 60 Gy, since 60 Gy delivered in fractions of 2 Gy comes down to exactly 30 fractions. Therefore, to compare different fractionation schemes, the EQD_2 is used.

Another treatment-related clinical feature is whether or not the treatment included Stereotactic Body Radiation Therapy (SBRT). SBRT is a cancer treatment that delivers a high dose to a small field of tissue. This generally results in less toxicity, as very little healthy tissue is irradiated (Timmerman et al., 2010). This type of treatment is often applied in case of a relatively small tumor, as small tumors can be mapped more precisely. However, if the tumor is located very close to more vulnerable healthy tissue, the high dose can result in very severe damage.

2.2.2 Dosimetrics

Dosimetrics or dosimetric features are input features taken from the dose-volume histogram (DVH). In turn, the DVH is constructed using the 3D dose distribution, on which the tumour and organs at risk are contoured. A DVH shows for each organ the volume fraction receiving at least a certain dose level radiation. A DVH is constructed as follows: each voxel (i.e. volume pixel, or 3D pixel) on the dose distribution has a value that represents the dose level in Gy in that particular location. For each dose level, the number of voxels with at least a certain dose level are counted and divided by the total number of voxels of the organ. This way, we can calculate the volume fraction of an organ receiving a dose of at least a certain amount of Gy. Dosimetrics are then taken from this DVH by looking at specific points in the histogram. Examples of dosimetrics are the normal lung V_{20} and the mean lung dose (MLD). The normal lung V_{20} is the fraction of the normal lung tissue receiving a dose of at least 20 Gy. The MLD is the mean dose that the the total volume of both lungs receives. The volume of healthy lung tissue receiving a certain dose level is used as the golden standard for predicting radiation-induced toxicity.

2.2.3 Radiomics

Radiomics or radiomic features are quantitative image features extracted from CT images and are able to capture characteristics from the image that are difficult to spot with the naked eye. For instance, certain tissue structures or textures may be related with a higher risk of radiotherapy related toxicities. There are multiple types of radiomics. As the name suggests, shape features are features that capture the shape of a segmentation (i.e. contour of organ). First order features are features taken from the first order histogram. Two types of second order radiomic features that are commonly used are taken from the gray-level co-occurrence matrix (GLCM) and the gray-level run length matrix (GLRLM). The first and second order features will be explained further below.

The grayscale in CT scans represents the Hounsfield unit (HU) of some tissue, describing its radiodensity. Different tissues have different HU values that can be distinguished on a CT scan. The first order histogram is a histogram where the relative frequency of each HU value (or grayscale value) is shown. The mean, standard deviation, skewness, or kurtosis are some examples of features that can be calculated from the first order histogram. Thus, these metrics are features that can be considered as a summary of the CT scan with its segmentations.

The GLCM looks at the gray-level of co-occurring voxels. It counts the amount of times a combination of two pixel values occur in the segmented part of the image. The matrix can be seen as a coordinate system, with the coordinates ranging from the minimum gray-level value to the maximum gray-level value. Let us look at an example. If two co-occurring pixels from the image have values of 1 and 2, the matrix will add 1 to the position in the matrix with coordinates (1,2). This is a good way of measuring the texture of a CT image and thus the texture of the lung tissue, which can be predictive of the development of toxicity after radiation. After creating the GLCM, various different features can be calculated from it. These features are similar in nature to the first order features calculated from the first order histogram. Some examples of second order features calculated from the GLCM are homogeneity, energy, and contrast.

The GLRLM is another way of measuring textures in lung tissue. In this matrix, the coordinates represent a gray-level value (x) and the number of times the particular value occurs in a row (y). Let us again look at an example. If there are in some image eight occurrences of three voxels in a row in one particular direction, all with a gray value of 5, the position in the matrix with coordinates (5,3) will have the value of 8. For each possible direction of the image, we can create a GLRLM. For 2D images this is four directions and for 3D images this is thirteen directions. In contrast to the GLCM, we now not only look for two co-occurring pixels, but rather for the length of runs that have the same gray value. The second order features are calculated from the GLRLM in a similar way to the second order features from the GLCM.

2.3 Classification Algorithms

Machine learning is a subfield of Artificial Intelligence, where we try to learn from data by applying algorithms to this data. We can then use these algorithms to make predictions on future events based on experience from previous data. A large number of classification algorithms have been developed. Three of the most well-known and evidence-based models that we used to predict acute radiation-induced toxicity are LR, RF, and SVM (Bishop, 2006). These classification algorithms are trained on labeled data and are therefore a form of supervised learning. In the following sections, the aforementioned algorithms will be further elaborated on.

2.3.1 Logistic Regression

LR is a regression algorithm that learns from data to make predictions for newly encountered individuals. LR makes use of a logistic or sigmoid function. Due to the nature of this function, LR can be used for binary classification:

$$h_\theta(z) = \frac{1}{1 + e^{-z^{(i)}}} \tag{2}$$

We call this the hypothesis. $h_\theta(x)$ is a value between 0 and 1, representing the probability of the outcome being positive. Using the discrimination threshold of 0.5, we can interpret this probability such that for all cases where $h_\theta(x) \geq 0.5$, we predict a positive outcome and for all cases where $h_\theta(x) < 0.5$, we predict a negative outcome. The variable $z^{(i)}$ represents the set of k weighted input features of individual i:

$$z^{(i)} = \beta_0 + \beta_1 x_1^{(i)} + \beta_2 x_2^{(i)} + \ldots + \beta_k x_k^{(i)} \tag{3}$$

Every $x_j^{(i)}$ is a feature of individual i. Every β_j is the weight that is assigned to input feature j through training on a data set and β_0 is the intercept. There are different ways of training your LR model, but the basic principle is often the same. First, the weights are randomly initialized. This will likely create a sigmoid graph that is a bad fit for the data set. We can calculate how bad of a fit this actually is using a loss function. The next step is to slightly alter the weights in such a way that the loss is reduced. We repeat this process until the loss function is minimized and we have found the weights for the optimal fit of our function. This process is called gradient descent. Many implementations of LR make use of a slightly altered version of gradient descent for a more efficient optimization process.

2.3.2 Random Forest

Before we get into RF, we will first discuss Decision Trees (DTs). Similar to LR, DTs can be used for classification or prediction. A DT consists of internal nodes and leaf (or bottom) nodes. Every node has a threshold value of a particular input feature and every leaf node (in case of binary outcome) represents either the positive or negative outcome. To classify a new individual, we start at the root (or top) node of the DT. Like every other internal node, the root node has a threshold for a particular feature. Depending on whether the feature value of the individual is below or above this threshold, we will traverse left or right down the tree. The rest of the tree is traversed in a similar way until we reach a leaf node, which then states the outcome of the individual.

The first step of growing a DT is selecting what feature to split on in the root node. We do this by calculating the average impurity of its child nodes from every possible split. A perfectly pure node only has either positive or negative individuals. If this is the case, we do not need to make another split, thus making it a leaf node. After calculating all average impurities of the potential child nodes, we select the split with the lowest impurity value. If the lowest impurity value is higher than the impurity of its parent node, the split is giving us no new information about the classification. Then, the split is not performed and the parent node also becomes a leaf node. This process continues until we have reached all leaf nodes.

RF is a method where we construct a large number of DTs, each with their predicted outcome class. The class with the most 'votes' from the DTs, will be the predicted outcome of our RF model. Constructing the DTs is done through bootstrap aggregating (bagging), where we use bootstrapped data to create multiple DTs. Bootstrapped data means that the data instances are randomly sampled from the original data set and can also occur multiple times in the bootstrapped data (because of replacement). Constructing DTs this way works great for reducing the impact of noise in the data set and therefore decreasing the variance of the model.

Another method RF uses is what we call feature randomness, where every DT can be constructed using only a randomly selected subset of all features. This is done to achieve a larger variation in DTs and a lower correlation between DTs. For instance, when a particular feature is a strong predictor of the outcome, many DTs will include a split on this feature, making the DTs highly correlated. When randomly selecting features however, it is unlikely that this feature will be picked for every DT, resulting in less correlated DTs and thus less overfitting.

2.3.3 Support Vector Machine

We will discuss SVMs by using an example of a data set with two input features and a binary outcome. In an ideal situation, when we plot all individuals on a two-dimensional graph with one input feature on the x-axis and the other on the y-axis, the individuals of the two classes are divided into two clusters. The goal of an SVM is to find a line between the clusters that separates them perfectly, so that when we encounter a new individual, we can classify it depending on what side of the line it appears on. To achieve the best classification performance, we would like this separating line to be exactly in between the two clusters. The SVM does this by maximizing the margin, which is the distance from the separator to the nearest individual of each cluster. The idea is similar when the model uses more than two input features. When there are three input features, the boundary becomes a plane and when there are four or more input features used, it becomes a hyperplane.

However, in many cases, perfect separation by the boundary is not possible and also not preferable, as it often results in overfitting. Therefore, we would prefer more of a soft margin as opposed to a hard margin as defined above. The soft margin is found by setting the boundary in such a way that we allow some classification errors to occur. By doing this, we find a separator that is less sensitive to outliers, resulting in a higher bias and lower variance.

Another difficulty is that not in every data set, the classes can be divided with a linear separator. The solution to this is what is called the kernel trick. The n-dimensional data are transformed into higher-dimensional data using a kernel function. In this higher-dimensional feature space, we can find a (linear) hyperplane separator, the same way as described before. Moving back to the original feature space, however, this hyperplane is now a non-linear separator. By using this method, we can create non-linear separators that are more accurate classifiers or predictors for many data sets. The SVM from scikit-learn (Pedregosa et al., 2011) that was used for this project has four different predefined kernel functions: a linear function (Equation 4), a polynomial function (Equation 5), a sigmoid function (Equation 6), and a radial basis function (Equation 7):

$$\langle x, x' \rangle \tag{4}$$

$$(\gamma \langle x, x' \rangle + r)^d \tag{5}$$

$$\tanh(\gamma \langle x, x' \rangle + r) \tag{6}$$

$$\exp(-\gamma \| x - x' \|^2) \tag{7}$$

In the polynomial function, the degree parameter d determines the degree of the polynomial. The parameter γ defines how far the influence of one training example reaches. A constant is added with the hyperparameter r.

2.4 Evaluation Strategies

To quantify the performance or usefulness of prediction models, several evaluation metrics are used in machine learning. They provide a good way of comparing different models and seeing which models perform best. Two evaluation metrics are discussed here: the Area Under Curve and the calibration curve.

2.4.1 Area Under Curve

One of the most widely used measures of performance of a prediction model is the Area Under Receiver Operating Characteristic Curve (AUROC) or Area Under Curve (AUC) for short (Bradley, 1997). As the name suggests, the AUC value is found by calculating the area under the ROC curve. The ROC curve plots the True Positive Rate (TPR) against the False Positive Rate (FPR) at varying discrimination thresholds. These values can be calculated from the confusion matrix for each threshold. A confusion matrix for a binary classification is a 2 × 2 matrix containing the number of True Positives (TP), False Positives (FP), True Negatives (TN), and False Negatives (FN) of a prediction. In these terms, positive and negative refer to what class was predicted for an instance, whereas true and false refer to whether the instance class is predicted correctly or incorrectly compared to the true data. We can measure these values against the total number of actual positive or negative instances to get their rate:

$$TPR = Sensitivity = \frac{TP}{TP + FN} = 1 - FNR \tag{8}$$

$$TNR = Specificity = \frac{TN}{TN + FP} = 1 - FPR \tag{9}$$

$$FPR = \frac{FP}{FP + TN} = 1 - TNR \tag{10}$$

$$FNR = \frac{FN}{FN + TP} = 1 - TPR \tag{11}$$

As can be seen from the formulas, TPR and TNR can also be called sensitivity and specificity, respectively. The x-axis of the graph is in practice often described as 1 − specificity, which is equal to the FPR. A discrimination threshold determines when an instance is considered to be positive or negative. When the predicted probability of an instance is above the threshold value, the instance is considered positive and when it is below the threshold value, the instance is considered negative. This means that we will have a different confusion matrix and thus a different TPR and FPR for each threshold.

A perfect classification model has an AUC score of 1 and if the model classifies everything incorrectly, the AUC value is 0. Randomly guessing will result in an AUC score around 0.5. The closer the value is to 1, the better the performance of the model. While the TPR and FPR alone can be misleading, the AUC shows a good representation of the discriminative ability of a model. The AUC score is in many cases preferred over the classification accuracy. This becomes more clear when we look at an example. When we are working with a skewed data set, where 95% of all individuals are positive and our classification model predicts all individuals to be positive, we have an accuracy of 0.95, but an AUC score of 0.5. Looking at just the accuracy, this seems to be a good model, but all it does is classify everything as positive. However, if we consider the AUC score, we find that this model will not be able discriminate very well on other, less skewed data sets. The AUC value takes into account the skewness of the data set and is therefore a better evaluation strategy for the classification model.

2.4.2 Calibration Curve

Another measurement of performance is a calibration curve. This curve shows the agreement between the observed and predicted outcome. The x-axis represents the predicted outcome, while the y-axis represents the observed outcome. The predictions are divided into equally large bins and for each bin, the average predicted probability and the actual fraction of positives are calculated. Ideally, the average predicted probability should be equal to the fraction of positives in each bin. For instance, if 80 out of 100 instances in a bin are positive, the perfect average predicted probability would be 0.8. Combining all bins from this ideal model would result in a calibration graph that is a straight diagonal line. We can now compare the calibration curve of other models to this straight diagonal line. When a calibration curve follows the diagonal line well, the model is calibrated well, but when it diverts from this diagonal line, we can see whether it over- or underestimates the outcome.

2.5 Problem

The problem studied in this project is the prediction of acute radiation-induced toxicity in lung cancer patients using only data that are available pre-treatment. Acute toxicity is defined as toxicity that occurs within the acute period of one week after treatment start till three months after treatment start. Moreover, we consider a patient to have experienced acute toxicity if the CTC grade of toxicity is at least 2. Predicting acute toxicity is useful to better create a suitable treatment plan for a specific patient. Ideally, oncologists would like to use a high enough dose to remove the tumor, but also to limit the damage done to surrounding healthy tissue. Dosimetrics such as the V_{20} and MLD, along with the patient's age and other clinical factors are already used by oncologists to improve on the treatment plan, but there are many more variables that might increase the chance of developing acute toxicity in patients. As it is impossible for oncologists to take all these factors into account, a machine learning approach might provide a better solution.

3 Related Work

There have been many studies that proposed methods to predict radiation-induced toxicity before radiotherapy treatment. A comprehensive search was performed in the PubMed[2] online database, using a carefully composed search term (Appendix A). The search results were manually screened for inclusion in this review. A number of papers were found that used blood biomarkers as input for their model (Spencer, Almiron Bonnin, Deasy, Bradley, & El Naqa, 2009; De Ruyck et al., 2011; Lee et al., 2015; Cui, Luo, Tseng, Ten Haken, & El Naqa, 2019; Yu et al., 2019). These papers were excluded from this review, because we do not have these data available for our project. In addition, some studies that were referred to by studies found by the search term were later included in this review. The results from this search strategy are reviewed in chronological order, to get a clearer understanding of the advances through time.

Munley et al. (1999) were the first to use an Artificial Neural Network (ANN) to predict radiation-induced toxicity. Unfortunately, we did not get access to the full article, hence it will not be included in this review.

Su et al. (2005) proposed a method of predicting the binary outcome of RP or non-RP using an ANN. The input that they used for their model consisted of dose-volume data (dosimetrics) in the form of a dose-volume vector V_D. The elements of this vector were the patient's lung subvolumes sampled from the individual patient's lung DVH. Using these data, three ANNs were trained and tested in a different manner. The first network (ANN_1) was trained by using the leave-one-out method, ANN_2 was trained a randomly selected 2/3 of the data and tested on the remaining 1/3, while ANN_3 was trained on a user selected 2/3 of the data and tested on the remaining 1/3. The leave-one-out method is where one patient's data are used to test the model that was trained on all of the remaining patients' data. This process is repeated as many times as there are patients in the data set. The data for ANN_3 were selected in such a way that there were as many cases with RP covering a large range of DVH distributions in the training set as possible. These different methods are an attempt to improve the quality of the training data. This is especially necessary when the quantity of data is lacking. The data set that was used consisted of 142 patients, of which only 26 suffered from RP. ANN_1 and ANN_3 performed well, with AUCs of 0.85 and 0.81, respectively, while ANN_2 performed significantly worse, with an AUC of 0.61. The sensitivity of ANN_2 and ANN_3, however, were low compared to their specificity. This is mainly due to the small number of RP cases in the data. This study shows that the quality of the data is very important for the performance of the model, but we expect that the difference in performance between these models will decrease as the quantity of data increases.

A group of authors has published five studies with different methods of predicting RP grade $\geq$ 2. The grading of RP in these studies is not done according to any version of CTCAE, but the grading system that they used is very similar. The classification algorithms that they used were: DTs, ANNs, SVMs, and self-organizing maps (SOM). Then, in a concluding paper, they combined all four of these methods to create an even more powerful prediction model. In the next paragraphs, we will go over all of these papers in order of publication.

In the first of the five studies they published, DTs were used to predict RP grade $\geq$ 2 (Das et al., 2007). The model that they have created seeks to improve the predictive capability of the commonly used dose-based parametric Lyman normal tissue complication probability (NTCP) metric. The NTCP quantifies the probability of injury for a given uniform dose to an organ. They do this by combining the NTCP metric with DTs through a process called boosting, using the AdaBoost algorithm. The combined model consists of a sequence of weighted individual decision

[2] `https://www.ncbi.nlm.nih.gov/pubmed/`

prediction units. The AdaBoost algorithm attempts to sequentially increase the model's predictive capability ("boost") by adding on each successive predictive unit, consisting of the sum of the NTCP metric and a DT. The DTs have dose and non-dose variables as input features. With the term dose variables, the authors refer to what we now know as dosimetrics. Examples of non-dose variables are the age and sex of the patient, tumor location, and type of treatment. The model was tested on a data set of 234 patients using 10-fold cross-validation. After 11 predictive units, very little changed in the AUC value, resulting in the optimal model consisting of 11 units. The AUC of this model was 0.72, which is a significant improvement over the AUC of 0.63 that the NTCP metric alone achieved.

In the second study that they published, they developed ANNs for the prediction of RP grade ≥ 2 (Chen, Zhou, Zhang, et al., 2007). Their goal was to improve on some of the limitations from the previous work with ANNs from Munley et al. (1999) and Su et al. (2005), as discussed earlier in this section. These limitations include the fixed number of nodes in the hidden layer, the lack of a selection process for input features, and that the input features did not include non-dose variables. To achieve a variable amount of hidden nodes, a growing/pruning algorithm was used, where first a very simple network is initialized, that is then alternately grown and pruned until a satisfactory solution is found. The input of the model consisted of features selected from a collection of 66 dose and 27 non-dose variables. The same growing and pruning strategy was used for the feature selection, where now, instead of the hidden nodes, the input nodes that represented input features were alternately grown and pruned until a satisfactory solution was found. The network was tested using 10-fold cross-validation. To investigate the addition of non-dose variables, two models were compared: $\mathrm{NN}_{\mathrm{dose}}$, using only dose variables, and $\mathrm{NN}_{\mathrm{all}}$, using both dose and non-dose variables. The performance of model $\mathrm{NN}_{\mathrm{all}}$ (after optimization through growing and pruning) showed an improvement over model $\mathrm{NN}_{\mathrm{dose}}$, with AUCs of 0.76 and 0.67 for models $\mathrm{NN}_{\mathrm{all}}$ and $\mathrm{NN}_{\mathrm{dose}}$, respectively. These results seem worse than results found by Su et al. (2005), but this can be mainly attributed to the more comprehensive testing methodology in this study and the larger data set of 235 patients.

In the third study that they published, SVMs were used to predict RP grade ≥ 2 (Chen, Zhou, Yin, Marks, & Das, 2007a). The setup of this study is very similar to the previous, except this time, instead of using an ANN, they used an SVM. The same 66 dose and 27 non-dose features were used as input, but a slightly different feature selection algorithm was used. Each feature is now only used as input if the AUC of the model would increase as a result of adding this particular feature. An input feature can also be substituted by another if this increases the AUC. This makes it quite similar to the growing and pruning method. In addition, when there are less than three input features, substitution cannot take place. The same 10-fold cross-validation was performed to train the model. Again, two models are compared: $\mathrm{SVM}_{\mathrm{all}}$ and $\mathrm{SVM}_{\mathrm{dose}}$. The AUCs of these models are similar to the ANN models, with values of 0.76 and 0.71, respectively, suggesting that the addition of non-dose parameters can improve discriminatory power of the SVM model. The data set that they used consisted of the same 235 patients as in the previous study.

In the fourth study that they published, they used SOMs to predict RP grade ≥ 2 (Chen, Zhou, Yin, Marks, & Das, 2007b). A SOM is a type of ANN that is trained using unsupervised learning. It is beneficial for reducing high-dimensional data to low-dimensional (often two-dimensional) data in the form of an easy to interpret map. It can be used to cluster patients based on similarities in their data, without taking the patient's outcome (RP or no RP) into consideration. Two patients with similar features will be situated in close proximity of each other on the map, with the Euclidean distance representing their similarity. The probability of a new patient outside of the data set getting RP after radiotherapy is associated with the distance to the closest patient on the map. SOMs are capable of finding synergistic interactions between the input features that can have very strong predicting power. The setup is, again, very similar to the other three studies described above. The input features consisted of the same 66 dose and 27 non-dose features. Two models $\mathrm{SOM}_{\mathrm{all}}$ and $\mathrm{SOM}_{\mathrm{dose}}$ were both trained and tested via 10-fold cross-validation and their performance was compared. The same data set of 235 patients was used. To select the most predictive input features, a similar strategy as in their previous works was used. The features could either be added or substitute a previously added feature if this increased the AUC of the model. The AUC of the optimized models $\mathrm{SOM}_{\mathrm{all}}$ and $\mathrm{SOM}_{\mathrm{dose}}$ were found to be 0.73 and 0.67, respectively. This, again, shows that the addition of non-dose input features results in an improvement in predictive ability of the model.

Das et al. (2008) then combined the models from the four previously described studies with the goal of improving the overall RP ≥ 2 prediction performance. They do so by fusing the models. The fusion is simply an average of all the predictions of the separate models. It was found that the fusion of all four models yielded an AUC of 0.79. Despite the performance of this approach being better than the separate models, using a fusion of four models is not optimal. The main disadvantage of this approach is that the fusion of four models is not easily interpretable, because one would have to examine all four models separately. In addition, training of all four models is very computationally intensive.

Gayou et al. (2008) proposed a method to predict RP grade ≥ 2 by using a LR model with a Genetic Algorithm (GA) for feature selection. They used the same alternative grading system that the previously discussed studies used. The input features that they used were dosimetric, clinical, physiological, and biological features. Using a GA is a very simple way of selecting what features to use as input for your model. Initialization is done by creating a population of sets of randomly selected input features (individuals). Every individual's set is then used as input for the LR model and the individuals are ranked by their performance measured by the fitness function. This fitness function is calculated using the performance of the model and the statistical significance of features in the individual's set. Individuals that rank the highest are more likely to be chosen as parents for the next generation. The next generation is generated by means of crossover between two parents and random mutations of genes (features). This process ends when the individuals converge to an optimal set of features, or when the maximum number of generations set by the user has been reached. This process was repeated for different sizes of feature sets of an individual, ranging from two to six features per set. This method of selecting features is compared to the brute force way of trying every possible combination of features as input for the LR model and then picking the best one according to the same fitness function. For the sets of up to five features, the GA selected the same features as the brute force method. Oddly enough, for the sets with six features, the GA selected features that showed a higher AUC value than the brute force method. According to the authors, this is due to the brute force method also using the fitness function to pick the best set of features, resulting in the GA selecting features that are more statistically significant. A GA can be a good way of selection features for any type of model, but given enough time and computational power, the GA should always perform as well as or worse than the brute force method on the test set. It can, however, prevent from overfitting on the test set, as it will not always select the theoretical best set of features for the test set.

With all the dose and non-dose variables, it is difficult to find the best predictors for RP grade ≥ 2 (CTCAE v4.0). Valdes, Solberg, Heskel, Ungar, and Simone II (2016) performed a univariable and multivariable analysis to find the best predictors and their threshold out of the abundance of available features. The algorithm they used for the univariable analysis was decision stumps, which is simply a DT with only one internal node and thus only one threshold for one feature. The multivariable analysis was performed to investigate the interaction between features. This was done by using DTs and two ensemble methods: boosting using the RUSBoost algorithm and RF. The difference between these algorithms is mainly that DTs are a weaker, but more easily interpretable classifier, while RUSBoost and RF combine the outputs of many weak classifiers to achieve a more powerful prediction, at the cost of being more difficult to interpret. More specifically, RUSBoost works well on skewed data sets, such as the one used in this work, while RF is useful for high variance problems, where DTs tend to overfit. It was found that on their data set of 201 consecutive stage I Non-Small Cell Lung Cancer (NSCLC) patients, there was no significant difference in performance between DTs and RF. RUSBoost did very slightly outperform both DTs and RF, but given its better interpretability, DTs might be the best choice for this problem. It is expected, however, that when more data are available, RF and RUSBoost will significantly outperform DTs.

Huang et al. (2017) proposed a method to predict acute esophagitis (AE) grade ≥ 2 (CTCAE v4.0). The purpose of this study was to test their previously published two-variable model on a new data set, to update this model, and to propose a novel machine learning-based predictive model. Only clinical and dosimetric features were used as input for the models. The features with low univariable correlations were filtered out before training the machine learning-based model. The features were then entered into a LR model with forward variable selection on a bootstrapped data set with replacement. Forward variable selection is a feature selection method that adds new variables one by one that give the best improvement to the model. The output of the LR model is then used as input for a least shrinkage and selection operator (LASSO) regression model, whose output is used to assess correlation with risk of AE. The predictive power of this model, as reported by the paper, is reasonable, but there are no real measures presented to support this claim.

Krafft et al. (2018) attempted to apply the idea of CT radiomic features to the prediction of RP grade ≥ 3 (CTCAE v3.0). In this work, the goal was to investigate the potential gain in predictive performance of adding pre-treatment total lung CT radiomics to the set of already tested (clinical and dosimetric) features. A total of 6538 radiomic features were divided into six different classes: 346 first order features, 5175 GLCM features, 825 GLRLM features, 50 Neighborhood gray tone difference features, 120 Laws' filtered features, and 23 Lung-specific features. LASSO LR was used to construct the prediction models, which also takes care of the feature selection. The different types of features were added step-wise to demonstrate the difference in performance between them:

1. Clinical variables (15)
2. Add 149 total lung dosimetric variables (164)
3. Add 149 heart dosimetric variables (313)
4. Add 6538 total lung image (radiomic) features (6851)

Using these features, the performance of the models tested on a data set of 192 patients measured by the AUC value in the same order (1-4) was: 0.47, 0.47, 0.51, and 0.68. The AUC values were averaged from 500 models total, resulting from 50 iterations of 10-fold cross-validation. The inner cross-validation loop was used to train the LASSO regularization strength hyperparameter. There is a significant increase when the radiomics are added to the equation. Seven radiomic features were present in at least 200 models (40%). No particular class of features, as described above, stood out in predictive performance. They expect that the addition of nonlinear modeling methods and larger data sets will likely improve the prediction of RP. It is demonstrated, however, that radiomics can be of great predictive power for the prediction of RP.

Bousabarah et al. (2019) created a prediction model for multiple outcomes, defined as local control (LC), disease-free survival (DFS), overall survival (OS), and development of local lung injury up to fibrosis (LF). LF is a complication that usually occurs three to six months after the last treatment. This means that in most cases, it would not fall under our definition of acute toxicity.

The symptoms are damaged and scarred lung tissue, which makes it more difficult for your lungs to work properly. Similar to the previous study, clinical, dosimetric, and radiomic features were used to predict the radiotherapy outcome. A total of 803 radiomic features were determined from the pre-treatment planning CT scan, consisting of first order features and GLCM features. Feature selection was performed using univariable Cox regression and multivariable LASSO regularization. Performance was measured by calculating the concordance score (c-index), which gives the probability of a randomly selected patient experiencing an event having a higher risk score than a patient who had not experienced the event. The model predicting LF had the highest c-index of 0.72 of all models predicting different radiotherapy outcomes. It was found that for LF, only first order features were found to be predictive.

Luna et al. (2019) published a study very similar to the study by Valdes et al. (2016) that was discussed earlier. The main difference between them is that in this study, the data set consisted of 203 stage II-III NSCLC patients, whereas a data set with stage I NSCLC patients was used in the previous study. Also, a different collection of clinical and dosimetric features was used. A similar univariable analysis was performed using decision stumps. Multivariable analysis was now only performed using RF. The performance of RF was compared to a number of different models: two other tree-based algorithms RUSBoost and CART, LR, and a linear SVM. The AUC values show that the RF model performed the best with an AUC of 0.66, followed by the linear SVM with an AUC of 0.65, LR with an AUC of 0.64, and RUSBoost and CART both with AUC scores of 0.63. The difference between the different methods, however, is minimal. Limitations of this study, according to the authors, are mainly the rather small size of the data set and the limited variation of input features that was used. Their suggestion is to extend the input features with more valuable information, such as PET/CT radiomic features.

Liang et al. (2019) then attempted to predict RP grade ≥ 2 (CTCAE v3.0) using dosiomics as input for their prediction model. Where radiomics take spatial features from radiological images such as CT scans, dosiomics take spatial features from the dose distribution image. The dosiomics were derived from the GLCM and GLRLM. A univariable and multivariable analysis was performed using LR. The models were trained and tested on a data set of 70 NSCLC patients. To compensate for the rather small size of the data set, it is bootstrapped 1,000 times and the resulting 1,000 bootstrap samples are used as training data sets for both the univariable and multivariable LR. The multivariable LR only searched for the best combination of two features. The optimal feature (univariable) or combination of two features (multivariable) were determined by the maximal mean training AUC of the bootstrapped data. In the multivariable analysis, a Spearman test was performed to exclude combinations of two features that were highly correlated with each other to prevent overfitting. After the feature selection process, the model was evaluated on the original data set. For both univariable and multivariable, three LR models with different types of input were compared: dosimetric features, Lyman NTCP features, and dosiomic features. Note here that the multivariable models thus did not make combinations of different types of input. The AUC values of the univariable models with the three types of input were 0.665, 0.710, and 0.709, respectively, while the multivariable models showed AUC values of 0.676, 0.744, and 0.782. This shows that in univariable analysis, NTCP features and dosiomics perform better than dosimetric features, but there is no clear difference in performance between them. In multivariable analysis, the dosiomics significantly outperformed both dosimetrics and NTCP features. The main limitation of this study is, however, the rather small size of the data set. To confirm the drawn conclusion, this method should be tested on a larger data set in the future.

The same authors came up with another approach to the same problem (Liang et al., 2020). The idea was to, instead of extracting features from the GLCM and GLRLM, apply a convolutional 3D neural network (C3D) to the same dose distribution images. Convolutional Neural Networks

(CNN) have so-called convolutional layers, in which a filter or convolution operation is used to extract features from images. A filter is a set of multipliers that go over every pixel of an image and take pixel values of surrounding pixels to calculate a new value. Depending on the filter values, a particular feature of the image is enhanced. The convolutional filters are parameters of the CNN that are at first randomly initialized. Similarly to any other parameter of the neural network, they can be learned through training of the network. When training is finished, the CNN has optimized the convolutional filters, so that it is able to detect patterns in the image that are predictive of the outcome. A CNN should be able to extract very subtle features from the images, that were otherwise neglected by using only dosiomic features. The convolution and pooling operations were similar to a regular CNN, but were now in 3D. The C3D network was pre-trained using a video data set for the task of action recognition to optimize the parameters. This was done because training on the small data set of only 70 NSCLC patients would not suffice. Guided gradient-weighted class activation map (grad-CAM) was used to map the regions of the dose distribution that were strongly correlated with RP. This method was compared to the three different multivariable models described in the previous paragraph. The C3D network significantly outperformed all other models with an AUC of 0.842. Although these results look promising, the same limitation of the previous study presents itself, in that the data set is very small and this method should therefore also be tested on a larger data set in the future in order to prove its validity.

4 Methods

The main steps of this project consist of preprocessing the data set, training the prediction models and evaluating the results. First, the raw data sets are described (section 4.1), after which we will go over the preprocessing steps in detail (section 4.2). A more general overview of this process including the initial covariate data sets, the data preprocessing and merging steps, and the final processed data set is shown in the flowchart in Figure 2. Every step of this flowchart is described in more detail in the following sections. Then, the outcome classes are described (section 4.3) and finally, the training of the prediction models is explained (section 4.4), along with its practical implementation (section 4.5).

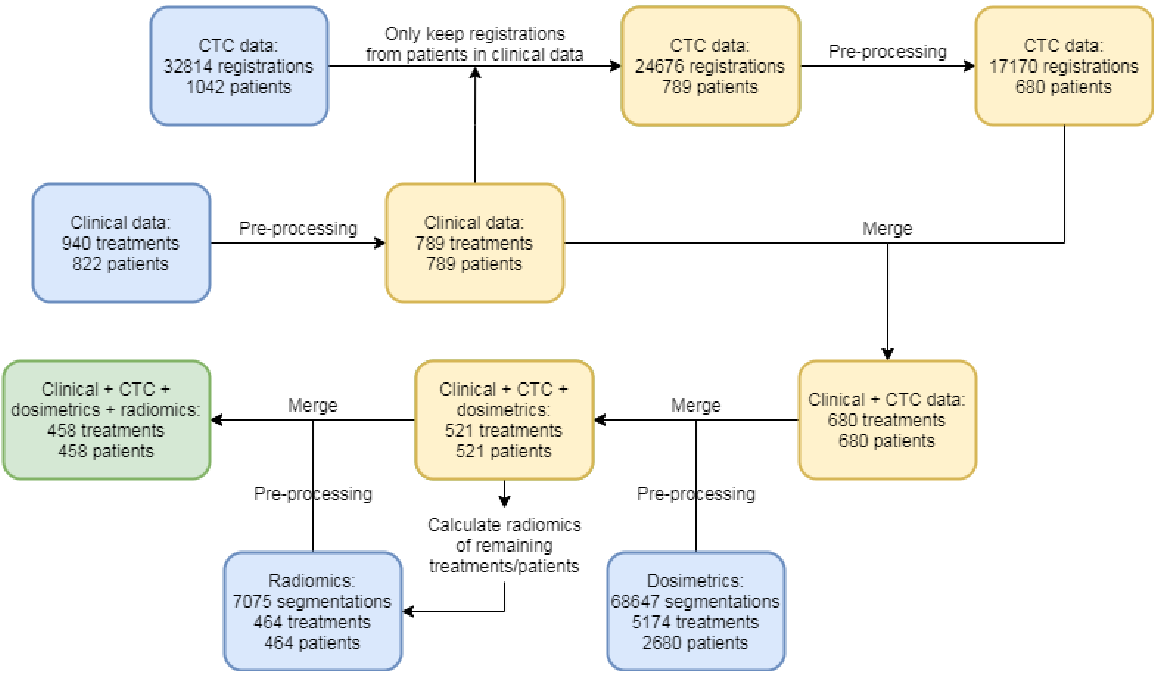


Figure 2: Flowchart showing the initial covariate data sets, the preprocessing and merging steps, and the final preprocessed data set. Blue = raw covariate sets, yellow = intermediate steps, green = final data set.

4.1 Data Sets

The final data set is a combination of four covariate sets, each containing a different type of information. The four covariate data sets consist of (1) clinical information of the patient and the treatment plan, (2) CTC toxicity registrations, (3) dosimetrics, and (4) radiomics. Apart from these four main covariate data sets, one other data set was used to merge the data sets together by adding some metadata. This is further explained in section 4.2.3. Moreover, some missing data were looked up in the patients' Electronic Health Record (EHR) and added to the data set. This is explained in section 4.2, whenever this was the case. Below, the patient cohort is described, after which all four raw covariate data sets before preprocessing are described.

4.1.1 Patient Cohort

The data set is a single institute cohort from the UMC Utrecht. All patients were referred to the UMC Utrecht from different hospitals in the region. The cohort consists of stage I-IV lung cancer patients that are treated with radiotherapy between 2009 and 2019.

4.1.2 Clinical Data Set

In the raw clinical data set, each row represents a treatment plan, along with the clinical information of the patient at the time of developing the plan. Initially, this data set consisted of 940 treatments, divided over 822 patients, where some patients were treated multiple times. Unfortunately, because this data set was generated from manual registrations by the radiation oncologist, some columns are incomplete and therefore not usable. In addition, many columns are simply not useful for the prediction of toxicity. From this covariate data set, we extracted nine features, of which four are features related to the patient and five are related to the treatment. All features and their values after preprocessing are presented in Table 1. The process of extracting these features from the raw data set is explained in section 4.2.1.

4.1.3 CTC Toxicity Registration Data Set

The raw covariate data set with CTC toxicity registrations are structured such that each row represents a CTC registration of one type of toxicity with a CTC grade, reported at a certain point in time by the patient. All 87 different types of CTC registrations are found in Appendix B. Table 2 shows the most relevant and frequent CTC registration types, along with the class they belong to and their frequency in the data set after preprocessing. The CTC registrations in the raw data set can be from any time before or after the treatment. There are many CTC registrations of different toxicities for each patient, often reported at multiple points in time. The raw data set consisted of a total of 32814 CTC registrations divided over 1042 patients. The process of extracting the acute toxicity for each patient from this raw data set is explained in section 4.2.2.

4.1.4 Dosimetrics Data Set

In the raw dosimetrics covariate data set, each row consists of 201 dosimetrics of one segmentation from the dose distribution of a treatment for some patient. A segmentation can be the contouring of an organ, the tumor or anything else that is relevant. There can be multiple treatments for one patient. This means that there can be, for instance, 30 rows for one patient, that include two treatments, each consisting of 15 segmentations. All 201 dosimetrics are described in Table 3. The raw data set consisted of a total of 68647 segmentations from 5174 treatments, divided over 2680 patients. The preprocessing of this raw data set is described in section 4.2.3.

4.1.5 Radiomics Data Set

The radiomics were calculated for the treatments remaining in the data set after preprocessing and merging the clinical features, acute toxicity labels, and dosimetrics together. A total of 32 radiomic features were used, of which 14 are shape features and 18 are first order histogram features. The complete list can be found in Table 4. The radiomics data set is structured similarly to the dosimetrics data set, where each row represents the radiomics for one segmentation from the planning CT scan of a treatment for some patient. The radiomics data set consisted of 7075 segmentations from 464 treatments, divided over 464 patients. The preprocessing of this raw data set is described in section 4.2.4.

	Feature	**Values**
Patient-related	Age	38-91
	Sex	Male/female
	Morphology	Adenocarcinoma/large cell carcinoma/ squamous cell carcinoma/unknown
	Stage of cancer	I/II/III/IV/unknown
Treatment-related	Radiotherapy dose	16-66
	Number of fractions	2-33
	EQD_2	31.8-276
	SBRT	Yes/no
	Chemotherapy	Yes/no/unknown

Table 1: Clinical features and their values after preprocessing.

Class	**CTC Registration**	**Frequency (registrations)**		**Frequency (patients)**	
		Total	**Grade $\geq$ 2**	**Total**	**Grade $\geq$ 2**
Fatigue/Any toxicity	Fatigue	1153	248	458	132
Esophagus toxicity/ Any toxicity	Anorexia	1110	47	448	27
	Nausea	1108	19	448	11
	Vomiting	1103	2	446	1
	Weight loss	1085	24	436	11
	Dysphagia	1016	115	416	52
	Esophagitis	127	11	73	5
	Pain	127	23	73	17
	Dehydration	1	0	1	0
	Aspiration	1	0	1	0
Lung toxicity/ Any toxicity	Cough	1016	42	416	21
	Dyspnea	1014	83	415	46
	Pneumonitis	999	6	409	5
	Pulmonary fibrosis	26	0	20	0

Table 2: CTC registrations per class along with their total frequency, their frequency of having grade ≥ 2, the number of patients having the CTC registration at least once, and the number of patients having the CTC registration at least once with grade ≥ 2. Patients can have multiple CTC registrations of grade ≥ 2. All values are from the data set after preprocessing. All CTC registrations also belong to the *any toxicity* class.

Feature	Description	Count
Volume	Organ volume	1
For x in 5-70 in steps of 5:		
Vperc$x$	Percentage of total organ volume receiving at least x Gy	14
VMLx	mL of organ receiving at least x Gy	14
For x in 5-100 in steps of 5:		
DminPerc$x$	Minimum dose to the x% volume receiving the highest dose	20
DmeanPerc$x$	Mean dose to the x% volume receiving the highest dose	20
DmaxPerc$x$	Maximum dose to the x% volume receiving the highest dose	20
DsumPerc$x$	Sum dose to the x% volume receiving the highest dose	20
For x in 0.1, 0.3, 1, 5-100 in steps of 5:		
DminMLx	Minimum dose to x mL of organ receiving the highest dose	23
DmeanMLx	Mean dose to x mL of organ receiving the highest dose	23
DmaxMLx	Maximum dose to x mL of organ receiving the highest dose	23
DsumMLxx	Sum dose to x mL of organ receiving the highest dose	23
	Total	201

Table 3: Description of all dosimetrics used for this project. This comes down to a total of 201 dosimetrics per organ. The data set after preprocessing contains these dosimetrics for nine organs: aorta, left bronchus, right bronchus, esophagus, heart, left lung, right lung, spinal cord, and trachea.

	Features	
Shape features	Elongation	Flatness
	LeastAxisLength	MajorAxisLength
	Maximum2DDiameterColumn	Maximum2DDiameterRow
	Maximum2DDiameterSlice	Maximum3DDiameter
	MeshVolume	MinorAxisLength
	Sphericity	SurfaceArea
	SurfaceVolumeRatio	VoxelVolume
First order features	10Percentile	90Percentile
	Energy	Entropy
	InterquartileRange	Kurtosis
	Maximum	MeanAbsoluteDeviation
	Mean	Median
	Minimum	Range
	RobustMeanAbsoluteDeviation	RootMeanSquared
	Skewness	TotalEnergy
	Uniformity	Variance

Table 4: All radiomics used for this project. There are a total of 32 radiomics for each organ of which 14 are shape features and 18 are first order features. The data set after preprocessing contains these radiomics for nine organs: aorta, left bronchus, right bronchus, esophagus, heart, left lung, right lung, spinal cord, and trachea.

4.2 Preprocessing

Each of the four covariate data sets was preprocessed before merging with the others. The preprocessing and merging was performed in the order shown in Figure 2. However, this order is quite arbitrary and can be changed if necessary. In the case of the CTC toxicity registration data set, the raw data set was immediately reduced to only include patients remaining in the preprocessed clinical data set from the intermediate step before. Somewhat similarly, the radiomics in the raw radiomics data set were only calculated for the patients remaining in the merged data set from the intermediate step before. These steps were taken to simplify the preprocessing of both data sets. This preprocessing and merging is described below for each covariate data set, where every decision of excluding a group of patients is explained carefully.

4.2.1 Clinical Data Set

The first step in preprocessing the clinical data was adding the end date of the treatment, as this was missing in the initial data set. The end date was mainly used for merging the data sets, for evaluation purposes and for the extraction of useful information from the CTC registration data, which will be elaborated upon later. For 16 treatments, the end date was missing. These treatments were looked up manually in the EHR of the patients, where we found that for 12 of them, the reason for the missing end date was one of two things. The first reason was that the patient had deceased before finishing the planned treatment entirely, resulting in no follow-up and thus no end date. Patients that did not finish their treatment were excluded from the data set. The second reason was that the treatment was only palliative. In a palliative treatment, the patient only receives one fraction of 8 Gy. This means that the end date is the same as the start date, which in practice often resulted in a missing end date in the data set. Palliative treatments very rarely result in (acute) toxicity and were therefore excluded from the data set. Thus, these 12 patients of which the treatment end date was missing were excluded from the data set. On top of that, one more palliative treatment of which the end date was not missing was also excluded. For the remaining 4 of the 16 treatments, the reason behind the missing end date was not clear. The end dates of these patients were looked up manually in the EHR and added to the clinical data set.

The largest exclusion step was the the exclusion of second-, third-, and fourth-time treatments from the data set. The problem that comes with these types of treatments is that patients that are treated more than once are often more likely to develop toxicities due to cumulative dose. This does, however, depend very much on what treatment(s) the patient has received before. It is difficult to capture this information, which is why they are excluded from the data set altogether and thus only first-time treatments are kept in the data set. In this step, 118 treatments were excluded, making the number of patients equal to the number of treatments in the data set.

Another problem was that that the CTC registrations in our data set are linked to a patient rather than to one particular treatment. This makes it impossible to directly distinguish toxicities caused by different treatments of the same patient. In practice, this means that if a patient is treated a second time within the acute period, there is no way to tell if a toxicity is caused by the first, second or even both treatments. For this reason, all first-time treatments that were followed up by a subsequent treatment within the acute period of their first treatment were excluded from the data set. Another 11 treatments were excluded from the data set because of this.

There were 163 patients in the data set with a missing value for their stage of cancer. In the EHR of these patients, 113 values for their stage of cancer were found, which were added to the data set. For the remainder of 50 missing stage of cancer values, a separate category named ‘unknown’ was created, as to not unnecessarily exclude all of these patients entirely. In the initial data set, the stages of cancer were not only divided in the stages I, II, III, and IV, but also subdivided in A, B, and C. Because some of these subdivisions contained only very few patients, the subdivisions were removed altogether, leaving only the categories I, II, III, IV, and ‘unknown’.

The raw clinical data set contained a column about the morphology of the patient, but most of the morphology types only occurred less than ten times. Also, there were 62 patients with a missing value. These missing values and rarely occurring morphology types were changed to ’unknown’. The three morphology types that did occur enough times to be used for the prediction were adenocarcinoma, large cell carcinoma, and squamous cell carcinoma.

A column containing whether or not the treatment included SBRT was added to the data set. The information needed to create this column was retrieved from a column containing treatment protocol. The input of this column was a summary code of the treatment protocol. This means that when a patient received SBRT treatment, ‘SBRT’ was part of the input of this column. Therefore, this column was scanned for the substring ‘SBRT’ and the SBRT column was set to 1 or 0 depending on whether the substring was found or not.

The chemotherapy column was already present initially in the data set. There were, however, many missing values, which were set to a new category called ‘unknown’. The addition of chemotherapy often results in an increase in toxicity, hence this feature was used as input for the models.

The EQD_2 was calculated using the radiotherapy dose and number of fractions using Equation 1 described in section 2.2.1. Because the α/β value is different for every type of tissue that is irradiated, the average value of three is used. We filtered out all treatments where the EQD_2 was below 30 Gy, because a treatment with an EQD_2 that is this low is very unlikely to result in any type of toxicity. Only two treatments were excluded because of this. Another treatment was excluded due to a missing value for the number of fractions.

In the final preprocessing step of the clinical data set, all nine clinical features were checked for missing or nonsensical values. One row was found with a zero value for the number of fractions and therefore an infinite value for EQD_2. This row was excluded from the data set. Another row was found with a value of 200 for the number of fractions. 200 fractions is not a value that you would expect here, because the irradiation is never divided over this many fractions in practice, hence this row was also excluded. After all preprocessing steps, we were left with 789 treatments and thus also 789 patients in our clinical data set.

4.2.2 CTC Toxicity Registration Data Set

As stated above, the CTC registration data did not include any treatment specific data, hence the start and end date from the preprocessed clinical data were added to the CTC registration data. Due to this merging, the CTC registration data was reduced from the total of 1042 patients to the same 789 patients remaining in the clinical data set.

Because we want to find the acute toxicity, we filtered the data on the measurement date of each CTC registration. As stated before, the acute period is from one week after treatment start till three months after treatment start. Thus, CTC registrations within this period are used to determine a patient’s acute toxicity. All patients that only have CTC registrations before this period are excluded, because it is then impossible to know whether acute toxicity occurred. 90 patients were excluded for this reason.

An issue here was that we cannot know for certain that whenever a patient lacks CTC registrations in the acute period, they have actually experienced no acute toxicity. It could be that the patient did experience acute toxicity, but it simply was not registered. To overcome this problem, we looked for CTC registrations after the acute period. If a patient lacks CTC registrations in the acute period, but does have CTC registrations after the acute period, we can conclude that the patient has experienced no acute toxicity, because otherwise this would have been registered as well.

Thus, only patients that lack CTC registrations within and after the acute period were excluded. 109 patients would be excluded in this step, but they were all looked up manually in the EHR to see if some CTC registrations could be found from the acute period. CTC registrations of 90 of these patients were found and added to the data set. Thus, only the remaining 19 patients were excluded from the data set in this step.

The purpose of adding the CTC toxicity registration data was to extract the acute toxicity for every patient. As explained in section 4.1, each row in the data set represents one CTC registration. For the *any toxicity* class, the highest value of all types of CTC registrations within the acute period was used to decide whether the patients scored positive or negative for this outcome class. If the highest CTC grade is at least 2, the patient is considered to have experienced acute toxicity. The same goes for the other three outcome classes, except only the highest value for the types of CTC registrations belonging to each class is used. What CTC registrations belong to what class can be found in Table 2. Besides the CTC registrations in this table, all other CTC registrations stated in Appendix B fall under the *any toxicity* class for completeness. These are, however, all very infrequent and thus not very significant.

After preprocessing and merging with the clinical data, the CTC toxicity registration data consisted of 680 patients. The nine clinical features, along with some metadata from the clinical covariate data set were merged with the acute toxicity labels of the four output classes. The merging was done based on patient ID. Merging only on patient ID is sufficient here, because for every patient, we made sure that only their first time treatment is included in the data set.

4.2.3 Dosimetrics Data Set

An issue with the dosimetrics data set was that there were no sufficient metadata to merge it with the other covariate data sets. The only thing that was available in this data set was the patient ID, but, as discussed before, we would like to merge not only on the patients, but also on their specific treatment. In other words, there was no way to know which row in the dosimetrics data set was linked to what specific treatment in the other data sets. To be able to merge the dosimetrics with the clinical and CTC toxicity registration data, another data set was used containing information of the patients' CT scans. The dosimetrics data set could be linked to the correct treatment in this data set using the description of the treatment. The CT scan date was then added to the dosimetrics data set and because this date was also present in the clinical data, the dosimetrics could be merged with the clinical and CTC toxicity registration data. However, for many of the rows in the dosimetrics data set, it was impossible to find the CT date. Due to this, 2055 patients were excluded from the dosimetrics data set. This seems like a very big loss of patients, but most of these patients were not even present initially in the clinical and CTC toxicity registration data set or were already excluded from these data sets. The remainder of excluded patients from this step is a much smaller amount.

Every row of the dosimetrics data set represents the dosimetrics of one segmentation. The names of the segmentations are stored in a column. It turned out, however, that the raw data contained many different names for the same segmentation. Some frequently occurring issues were that the name was both capitalized and non-capitalized, different languages, and different kinds of abbreviations. The capitalization issue is easily fixed, but to fix the other issues, as many different names as possible for one segmentation were searched. Also, to make sure that all completely different clinical names of a segmentation (e.g. myelum/spinal cord) were found, radiation oncologists were consulted. A good example of a segmentation with many different names is the left bronchus. This organ was found to have 15 different names. All different spellings of the same segmentation were changed such that they all fall under the same name.

Only the most relevant and most frequently occurring types of segmentations are included in the data set. Nine segmentations were by far the most frequently occurring, with at least double the amount of occurrences compared to the next most frequently occurring segmentation. These segmentations are all organs: aorta, left bronchus, right bronchus, esophagus, heart, left lung, right lung, spinal cord, and trachea. These organs are also the most relevant segmentations for the prediction of acute toxicity. Therefore, only patients of which all nine of these organ segmentations were available were included in the data set. Only 6 patients were excluded because all nine organ segmentation were missing and another 63 patients were excluded because at least one of the nine organ segmentations was missing.

To be able to merge the dosimetrics data set with the clinical features and acute toxicity labels, we need each row to contain all dosimetrics of one patient, instead of having a separate row for each segmentation. For each patient, all dosimetrics in each row, representing one of the nine organ segmentations mentioned above, were cast into one row. Now, every row consists of 201 dosimetrics for each of the nine organs, resulting in a total of 1809 features per patient.

After preprocessing the dosimetrics, they were merged with the clinical features and acute toxicity labels. After merging them together on the CT date and patient ID, 521 patients remained in the preprocessed data set.

4.2.4 Radiomics Data Set

As discussed in section 4.1.5, the radiomics were only calculated for the patients remaining in the data set after merging the clinical features, acute toxicity labels, and dosimetrics. Due to errors in the radiomics extraction process resulting from mismatching coordinate systems between the segmentation mask and CT scan, it was not possible to calculate the radiomics for all 521 remaining treatments. The radiomics were calculated for 461 of the 521 patients, excluding 60 patients in the process.

Many of the same preprocessing steps performed on the dosimetrics were also performed here. The issue of having many different names for the same segmentation occurred here as well, to which the solution that was applied in the dosimetrics data set was applied here as well. Also, only the segmentations of the same nine organs were included here and the features in multiple rows were cast in a single row for each patient as well. For three patients, one of the nine organ segmentations was missing, hence these three patients were excluded from the data set.

The radiomics were merged with the clinical features, acute toxicity labels, and dosimetrics on the ID of the patient and the specific treatment. Because all of the patients in the radiomics data were also in the merged data set, no extra patients were excluded due to the merging, resulting in a final data set of 458 patients.

4.3 Outcome Classes

The different outcome classes are already briefly discussed in section 4.1, but we will discuss them in more detail here. We decided to introduce the classes, because predicting one type of CTC toxicity at a time can be very troublesome. For instance, if we only consider the CTC registrations for RP, we would find that only 6 out of 458 patients would have a grade of at least 2. This distribution is very skewed, making it very difficult to predict this particular outcome. Predicting all types of CTC toxicities at once, as we are doing for the *any toxicity* class, might also be difficult. Because there is such a wide range of toxicities for this class, the underlying causes will likely vary even more. Nonetheless, it is interesting to look at this overarching class, because there will likely still be some relation between the input features and the outcome. Also, being able to predict any type of toxicity would be useful in practice. Fatigue is the CTC registration that was by far the most frequent in our data set. Moreover, fatigue is a toxicity that can be caused by anything, making it difficult to relate it to one particular organ. For these reasons, fatigue was made into a class of its own. The rest of the CTC registrations were examined and assigned to an organ where the toxicity likely originates from. All of the frequently occurring CTC registrations were assigned to the esophagus and lung. The outcome classes with the CTC registrations that belong to them are presented in Table 2. For the rest of the CTC registrations, only eight patients were registered with a grade ≥ 2, which is too few for another outcome class. These CTC registrations were considered under *any toxicity*, along with the CTC registrations of the other classes. The distribution of each class can be found in Table 5.

Class	**Yes**	**No**
Any toxicity	193	265
Fatigue	126	332
Esophagus toxicity	88	370
Lung toxicity	52	406

Table 5: Distribution of each class, where 'Yes' means scoring positively for our acute toxicity criterion and 'No' means scoring negatively. The total number of patients is 458. Patients can score positively for multiple classes.

Besides the different CTC registrations the classes are based on, there is another distinction between them. For each outcome class, we use dosimetrics and radiomics of different groups of organs. For the classes *any toxicity* and *fatigue*, we use all dosimetrics and radiomics of all nine organs, because these toxicity classes are not related to any organs specifically. The *esophagus toxicity* and *lung toxicity* class, however, are specifically related to certain organs. Therefore, we will only use dosimetrics and radiomics of the esophagus for the *esophagus toxicity* class, and dosimetrics and radiomics of both lungs and bronchi for the *lung toxicity* class.

4.4 Prediction Models

As discussed earlier, three different classification algorithms were used: LR, RF, and SVM, which are all trained using a nested 5-fold cross-validation. The cross-validation was chosen to be 5-fold to make sure enough patients with acute toxicity were present in each test set, while making sure the training sets do not become too small. Making the test set larger means that there is a smaller portion of the data set left for training, which increases the chance of underfitting our model. 5-fold cross-validation was found to be a good middle ground solution. LR and SVM both make use of Ridge Regularization. Multiple models were trained using every possible combination of the different input features, classification algorithms, and output classes. This was done in order to be able to find what types of input are the best predictors for each outcome class, what classification algorithm is best suited for this problem, and which outcome class we can predict the best. In this section, nested cross-validation will be explained in detail and the hyperparameters for each classification algorithm will be discussed.

4.4.1 Nested Cross-validation

Similarly to normal cross-validation, in nested cross-validation the data are divided into a number of folds, creating training and test sets. The difference with normal cross-validation is, however, that every training set is then divided again into a number of folds, creating inner training and test sets within the outer fold's training set (Krstajic, Buturovic, Leahy, & Thomas, 2014). The inner folds are used to optimize the hyperparameters via a grid search. Each hyperparameter has its own grid with values to try out. Models are trained on the inner training set trying out all combinations of hyperparameter values from the hyperparameter's grid and are then evaluated on the inner test set. The model with the best evaluation score on the inner test set is considered to have the best hyperparameters. Then, using the best hyperparameters for this outer fold, a model is trained on the outer training set and is then evaluated on the outer test set. This means that every outer fold uses a different set of optimal hyperparameters. The final AUC score is then calculated by comparing the joint predictions of all outer test sets with the true toxicity outcome of the data set. For both the inner and outer folds, stratified cross-validation is used to make sure that every fold contains roughly the same amount of positive and negative outcomes.

The benefit of using nested cross-validation over normal cross-validation is that different data are used to tune the hyperparameters and to evaluate the model. Without nested cross-validation, the quality of the hyperparameters are evaluated on each test set, which is also used for evaluation of the model. Doing this increases the chance of overfitting our model. Thus, nested cross-validation is used to overcome this bias.

4.4.2 Hyperparameters

The grid search in the nested cross-validation tries to optimize the hyperparameter values by minimizing the negative log-likelihood function on the inner test set. The model that achieves the lowest value for this loss function is considered to be the model with the best hyperparameters. The negative log-likelihood function is the standard criterion to fit models in statistics (Bishop, 2006).

Two things were done to make sure that each hyperparameter grid contains a good approximation of the optimal value. Firstly, after training the models, the optimal values that were found for each hyperparameter were compared to the hyperparameter's grid. When the optimal value of a hyperparameter is found to be the smallest or largest value of its grid, it could be that a better value can be found outside of the grid. If this was the case, the grid was either shifted or made larger to make sure a better approximation of the optimal value was present within the grid. Secondly, the grids were as densely filled as possible while staying within reasonable time of training the model, making use of the resources available. Having a more dense grid with a sufficiently wide range of values means that we get a better approximation of the optimal value.

Each classification algorithm requires a different set of hyperparameters to be tuned. Below, we will discuss the hyperparameters and the values that were included in the grid search for each classification algorithm separately. The optimal hyperparameter values that were found for each model are not mentioned, because there are too many. Every model consisting of a combination of an input type, classification model, and output class utilizes different optimal hyperparameter values. Moreover, due to the nested cross-validation, every model even utilizes different optimal hyperparameters for each outer fold. Averaging over the outer folds would have been a good idea, but this is in some cases not possible. For example, if an SVM model uses different kernel functions for some outer folds, different hyperparameters are optimized, over which we cannot average. Instead, the best performing model is discussed in detail in section 5.4, along with its optimal hyperparameter values.

4.4.2.1 Logistic Regression

For LR with Ridge regularization, there was only one hyperparameter to train which determines the regularization strength. In the implementation from scikit-learn, this is defined as the inverse regularization strength C, where the lower the value, the stronger the regularization. The grid for this hyperparameter consisted of a range of fifty values on a logarithmic scale from 1e−15 to 1e+4. The lower bound of this grid is very low, because for some types of input, a very strong regularization was found to be optimal.

4.4.2.2 Random Forest

Two hyperparameters were predefined and three hyperparameters were optimized for the RF classifier. The predefined hyperparameters were whether to enable bootstrapping and the number of trees in the forest. Bootstrapping was enabled to reduce variance of the model. Increasing the number of trees will always result in a stochastic decrease of the model's loss. This means that a larger number will always be better for the performance of the model. The cost of having more trees is that training the model will be much more computationally intensive. Thus, searching for the best value for this hyperparameter through grid search is not necessary. The number of trees in the forest was set to 1000, because after more than 1000 trees, the benefit of adding more trees becomes minimal.

The three hyperparameters that were optimized in the nested cross-validation are the maximum depth of the trees, the maximum number of features to be considered for each split in a tree, and the impurity measure to be used at each split. Depending on the type of input, the optimal maximum depth was overall found to be relatively low. Therefore, the grid was very dense around the lower values containing all values from one to five and values with increasingly larger steps up to a value of 80, to make sure all outliers are included. The maximum number of features could be only one of two things: the base-2 logarithm or the square root of the total number of features. Both are default values in scikit-learn and are generally considered to be good methods of selecting the maximum number of features. Higher values for this hyperparameter mean having a higher probability of selecting 'good' features, but will also reduce diversity between the trees and thus reduce the overall model's performance. The measures of calculating the impurity of a node are Gini and entropy. They are very similar ways of calculating the impurity, but can result in very subtle differences.

4.4.2.3 Support Vector Machine

The SVM has the most hyperparameters to tune of the three classification algorithms. The same inverse regularization strength hyperparameter C needs to be tuned for the Ridge regularization. Because this is not the only hyperparameter that needs to be tuned here, a much smaller grid is chosen than the one used for LR. It consists of a range of fifteen values on a logarithmic scale from 1e−12 to 1e+4. The most important hyperparameter for the SVM is the kernel function, which, as discussed in section 2.3.3, has four options in scikit-learn: a linear function (Equation 4), polynomial function (Equation 5), sigmoid function (Equation 6), and radial basis function (Equation 7). Each kernel has different parameters that require tuning. These parameters are treated as any other hyperparameter in our model.

For the polynomial function, the three hyperparameters d, γ, and r need to be tuned. We can see from the equation that if we set the degree to 1, we have a linear function. Because the grid search of the polynomial function includes this degree value, the linear function itself was not included in the grid search. Through testing it was found that in cases where the polynomial function performed the best, the degree was often a number below four. For that reason, the grid contained all values up to three. For completeness, the grid contained values up to a highest value of ten, skipping some values to reduce training time. The hyperparameter γ could be one of two values: $1/(n\_features \times variance(X))$ or $1/n\_features$, where $n\_features$ stands for the number of features. These are the two default values in scikit-learn that are commonly used for this hyperparameter. The hyperparameter r simply adds a constant. As this contributes little to the model and in order to reduce the complexity of the model, this hyperparameter was left at its default value of zero.

From the equation of the sigmoid function, we can see that it has the same hyperparameters as the polynomial function, except for the degree (d). The grid search for the sigmoid function includes the same values for γ and r as the polynomial function.

The radial basis function only has the hyperparameter γ. Again, the same grid for this hyperparameter was used for the grid search.

4.5 Implementation

Two different programming languages were used: R (R Core Team, 2020) and Python (Van Rossum & Drake, 2009). R was used to preprocess the data, while Python was used for the training and evaluation of the models. The data.table package in R (Dowle & Srinivasan, 2020) was used to store and manipulate data. The packages pandas (Wes McKinney, 2010), NumPy (Harris et al., 2020), and scikit-learn (Pedregosa et al., 2011) were used for the implementation of the models in Python. More specifically, the functions LogisticRegressionCV, RandomForestClassifier, and SVC from scikit-learn were used for the classification algorithms. GridSearchCV from scikit-learn was used to perform the grid search in the nested cross-validation. The results were plotted using matplotlib (Hunter, 2007).

The machine that the models were trained on contained an Intel Xeon Gold 6240 CPU @ 2.60GHz processor with 72 cores, of which 25 were available for use in this project. The RAM consisted of 263.59 GB total.

5 Results

The data set that the models were trained on contained the clinical features, dosimetrics, radiomics, and acute toxicity labels for 458 patients. Every combination of input type (7), classification algorithm (3), and outcome class (4) was evaluated, resulting in $7 \times 3 \times 4 = 84$ AUC values. The AUC results are plotted in a bar chart for each outcome class in Figure 3. For robustness, the results shown in the bar charts are averaged over ten runs. The difference in performance between the different outcome classes, classification algorithms, and input types is discussed here. In addition, we will look at the model with the highest AUC in more detail.

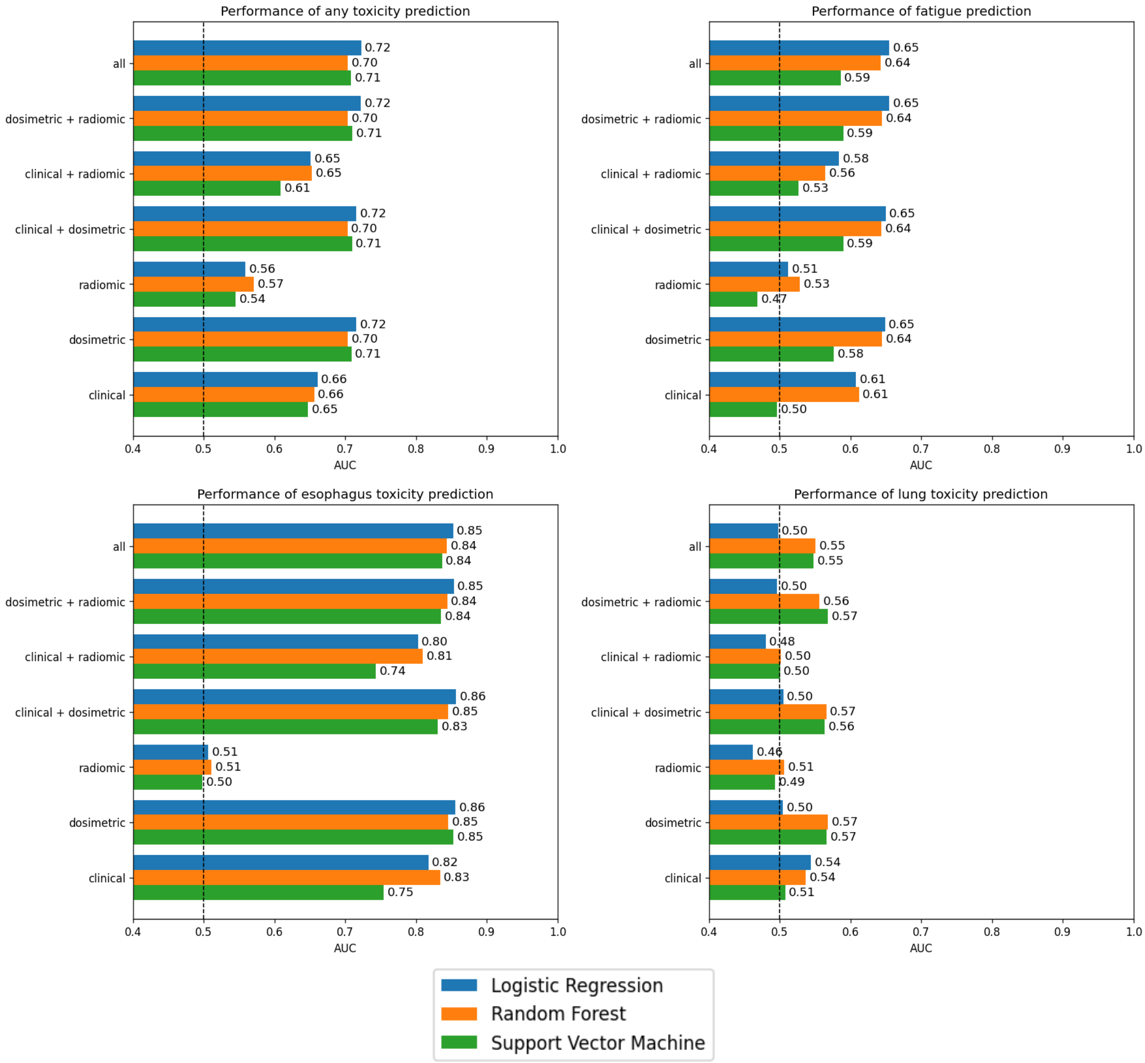


Figure 3: Performance of acute toxicity prediction for each class, with all combinations of classification algorithms and input types. The AUC values are averaged over ten runs.

5.1 Class Comparison

In general, we can see from the bar charts that the *esophagus toxicity* class is predicted the best with an AUC score of 0.86, followed by *any toxicity* with an AUC score of 0.72, *fatigue* with an AUC score of 0.65, and *lung toxicity* with an AUC score of 0.57. For all classes except *lung toxicity*, the best score was achieved by using a LR model with any combination of input types that includes dosimetrics. For the *lung toxicity* class, RF with any combination of input types that includes dosimetrics performed best.

The *any toxicity* class is made up of all toxicities from the other three classes. Therefore, it is not surprising that we see that the AUC score of predicting this class is around the average of the other classes. Even though it appears to predict above the average of the other classes, we have to keep in mind the number of patients having each type of toxicity (Table 5). Especially *lung toxicity* is a rather small class compared to the other classes and will therefore have less impact on the average score over all classes combined.

The *fatigue* class is a difficult one to predict. This is because being fatigued is very subjective. One person might feel fatigued much more quickly than another. Having any form of lung cancer is fatiguing to most patients, as the tumor eats up a lot of energy and it can often cause sleep deprivation. Also, fatigue can be caused by external factors outside of the radiotherapy much more than the other classes might be. One example of this is the time of appointment. It could be that a patient has to get up very early, causing them to be fatigued. However, there are still many features that might predict fatigue in a patient, such as their age or perhaps the number of fractions that cause the patient to have multiple appointments throughout one week. Taking all this in mind, the AUC score for this class is expected to be around this value.

The huge difference between predictability of esophagus and lung toxicity is more difficult to explain. The clinical and dosimetric features are very predictive of esophagus toxicity and especially LR captures this relation well. The models predicting lung toxicity perform poorly, indicating that the relation between the input and this particular kind of toxicity is less clear. An explanation for this could be that the dosimetrics and radiomics in our model are from the left and right lung separately, instead of the two lungs together as a whole. When a tumor located in the left lung is irradiated with a high dose, the dosimetrics for the right lung will not necessarily have high values, but the risk of having radiation-induced lung toxicity is high. Because of this issue, the relation between the dosimetrics and radiomics and the lung toxicity will be weaker. The same thing goes for the left and right bronchus, which are also included for the *lung toxicity* class. This issue does not, however, explain the bad performance in predicting *lung toxicity* using the only clinical features, because these features are not related to any organs specifically. It could be that for lung toxicity, only radiotherapy is important, which is only captured by the dosimetric features.

5.2 Classification Algorithm Comparison

As briefly mentioned in the previous section, LR is in many cases the best classification algorithm for this problem. We do see, however, that in some other cases, RF or SVM might be the best choice. In particular, the lung class is predicted much better overall using the RF and SVM classifier. Moreover, in some cases where clinical or radiomic features were used for training, RF performed better than LR. It is not clear why the latter is the case. There are almost no input types or outcome classes for which an SVM is the best choice.

Lung toxicity is the only class where RF and SVM perform better than LR for most types of input. The only type of input with which LR still performs the best is the clinical features. Using the radiomics as input, there is no real difference between the three classification algorithms, as they all perform badly. LR performing worse than RF and SVM when predicting *lung toxicity* using dosimetrics as input might also be explained by the fact that the dosimetrics are from both lungs and bronchi separately, as explained in the previous section (5.1). This issue can be dealt with by using the max operation on a dosimetric feature in order to find the highest feature value between the two lungs or bronchi. By doing this, a lung or bronchus with a low value for some dosimetric feature will not negatively affect the predicted outcome if the other lung or bronchus has a high value for this feature. This operation is, however, a nonlinear operation. This means that linear models, such as LR are unable to perform such an operation, resulting in a worse overall performance in this case. Nonlinear models, such as RF and SVM are able to do this, which explains why their performance is better for this outcome class and input type. The fact that LR does have the highest performance in predicting *lung toxicity* using only clinical features supports this idea, as these features are not related to any organs specifically and thus do not benefit from the added nonlinearity. Dosimetrics of both lungs and bronchi are also used as input for the *any toxicity* and *fatigue* classes, but they make up a much smaller portion of the total number of dosimetric features, as many more other organs are used as input as well.

5.3 Input Feature Comparison

Overall, the models trained on any combination of input types that includes dosimetrics achieve the highest AUC scores. Also, there is not much difference in performance between the combinations that include dosimetrics. Intuitively, it is not very surprising that every combination of input types that includes dosimetrics are the most predictive for every outcome class, as these are features that contain information about the radiotherapy treatment plan itself.

In general, the clinical features on their own are slightly less predictive than the dosimetrics. Clinical features can provide a very good and much simpler way of predicting acute toxicity, but they are not as predictive as dosimetrics.

The models trained on only the radiomics have an AUC score of around 0.5 for every class except *any toxicity*, where the AUC score for RF is 0.57. Models with an AUC score of 0.5 are equal to randomly guessing the outcome. It is remarkable that the *any toxicity* class can be predicted slightly better, but an AUC score of 0.57 is still not a very good result. The overall bad performance of models using radiomics as input shows that radiomic features from the CT scan before the treatment are less important for the prediction of acute toxicity.

Another thing that stands out is that the models using the combination of clinical and radiomic features perform slightly worse than the models using only clinical features. This could be due to the regularization. Because the radiomic features have such little predictive power and there are much more radiomic features than clinical features, the regularization strength will likely increase, causing all features to have less effect on the outcome. This means that the clinical features will have less effect on the outcome as well. This negative effect of radiomics on the prediction performance is smaller in models with input combinations that also include dosimetrics. This is because there are even more dosimetric features than radiomic features, causing the regularization strength to not increase as much.

5.4 Best Model

As discussed above, the LR models predicting *esophagus toxicity* using dosimetrics and the combination of dosimetrics and clinical features as input achieve the highest AUC scores of 0.86. In Figure 4, the ROC and calibration curve of the model using clincial and dosimetric features are presented. There is not much we can take from the ROC curve besides the area underneath it. Overall, the calibration curve follows the diagonal line well, meaning that the model is well calibrated. From the class distribution in Table 5, we can calculate that the fraction of patients that experienced acute toxicity related to the esophagus is equal to $88/458 \approx 0.192$. The calibration curve shows that six of the ten bins are below a fraction of positives and predicted probability of 0.192. This means that the model does not just predict the majority class in many cases, but can actually discriminate the patients without acute toxicity from the patients with acute toxicity well. The confusion matrix presented in Table 6 is created using a discrimination threshold of 0.5. From the matrix, we can see that the model does a good job of correctly predicting the true class.

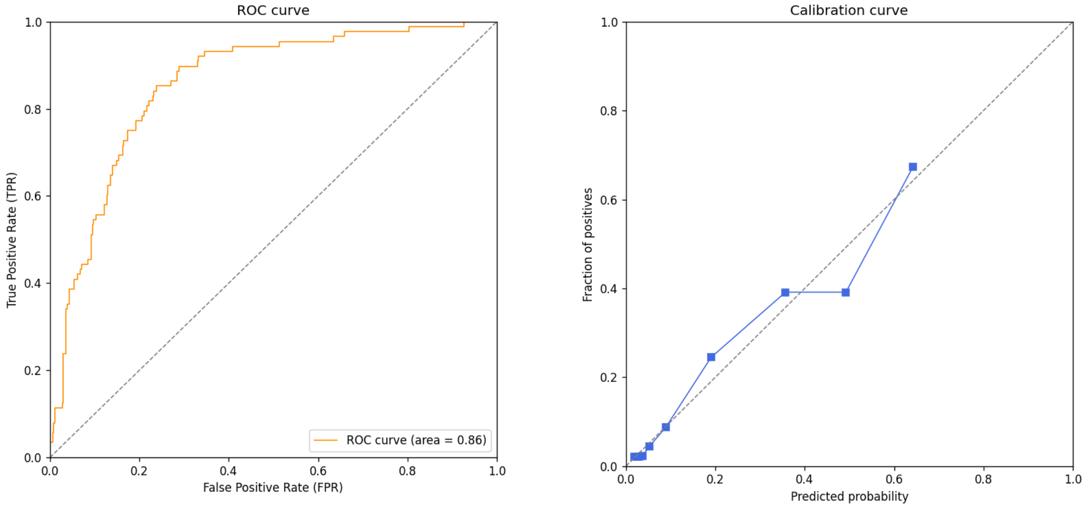


Figure 4: ROC and calibration curve for the best performing model: LR model predicting *esophagus toxicity* using dosimetric and clinical features. The calibration curve is divided into 10 bins.

		True	
		Yes	**No**
Predicted	**Yes**	39	28
	No	49	342

Table 6: Confusion matrix of the best performing model: LR model predicting *esophagus toxicity* using dosimetric and clinical features. The matrix is created using a discrimination threshold of 0.5.

The LR coefficients of the model predicting *esophagus toxicity* using dosimetric and clinical features are presented in Table 7. For comparison, the LR coefficients of the model predicting *esophagus toxicity* using only clinical features are presented in Table 8. The categorical clinical variables were converted to dummy variables for practical purposes and a full description of the dosimetric features can be found in Table 3. For both Tables, only the top fifteen dosimetric features with the highest coefficient values are presented. The inverse regularization strength C value of the clinical + dosimetric model is smaller than for the clinical model, with values of 9.856e−3 and 4.803e−2, respectively. This is as expected, because the clinical + dosimetric model has many more features in total, which increases the chance of including less predictive features, causing the regularization strength to increase. Due to the difference in regularization strength in each model, the coefficient values are not directly comparable between them, but the order in which they are ranked is interesting to look at.

Something that immediately stands out is the fact that the clinical features Number of fractions, SBRT, and EQD_2 are among the highest ranking features in the clinical model, while they are among the lowest ranking features in the clinical + dosimetric model. These are all treatment-related clinical features, which means that they might be captured partially by the dosimetrics as well. Therefore, it is not surprising to see that when we add dosimetrics to our model, these features become less predictive. We do see that in both models, the features *Age* and *Sex* are among the most predictive patient-related clinical features. The dosimetric features are more difficult to compare, as there are so many of them and their coefficients lie not very far apart. Therefore, it is difficult to draw hard conclusions from this. We do see that different variations of *DminPerc* and *VML*/ *Vperc* have some of the highest coefficient values, suggesting that these dosimetric features are the most predictive for acute esophagus toxicity.

Clinical feature	**Coefficient**	**Dosimetric feature**	**Coefficient**
Sex - female	0.1005	VML70 - esophagus	0.0364
Age	-0.098	DminPerc15 - esophagus	0.0345
Morphology - large cell	-0.0805	Vperc70 - esophagus	0.0321
Morphology - unknown	0.0566	DminPerc20 - esophagus	0.0316
Radiotherapy dose	-0.0465	DminPerc10 - esophagus	0.0306
Stage of cancer - III	-0.0452	Vperc65 - esophagus	0.0292
Stage of cancer - unknown	0.0436	DminML5.0 - esophagus	0.0291
Chemotherapy - unknown	-0.0394	DmeanPerc25 - esophagus	0.0269
Morphology - squamous cell	-0.0385	DmeanPerc20 - esophagus	0.0269
Chemotherapy - no	0.0357	DminPerc40 - esophagus	-0.0268
EQD_2	-0.0295	DminPerc50 - esophagus	-0.0250
Stage of cancer - IV	0.0288	VML55 - esophagus	0.0246
Stage of cancer - II	0.0215	DmeanPerc15 - esophagus	0.0245
Number of fractions	0.0091	DmeanPerc30 - esophagus	0.0242
SBRT	-0.0042	DmeanML5.0 - esophagus	0.0233

Table 7: LR coefficients of the best performing model: LR model predicting *esophagus toxicity* using clinical and dosimetric features. All clinical features that were used as input are presented here, along with the top fifteen dosimetric features with the highest coefficient values. The categorical variables were converted to dummy variables for practical purposes. For a full description of all dosimetric features, see Table 3. The average value for the inverse regularization strength C over the five outer folds is 9.856e−3.

Clinical feature	**Coefficient**
Number of fractions	0.379
SBRT	-0.366
EQD_2	-0.345
Age	-0.265
Sex - female	0.175
Radiotherapy dose	0.147
Stage of cancer - IV	0.135
Morphology - large cell	-0.106
Stage of cancer - III	0.089
Chemotherapy - no	-0.088
Morphology - unknown	0.059
Stage of cancer - unknown	0.056
Chemotherapy - unknown	-0.054
Morphology - squamous cell	-0.025
Stage of cancer - II	0.020

Table 8: LR coefficients of the model predicting *esophagus toxicity* using only clinical features. All clinical features that were used as input are presented here. The categorical variables were converted to dummy variables for practical purposes. The average value for the inverse regularization strength C over the five outer folds is 4.803e−2.

6 Discussion

Three classification algorithms were used to predict acute toxicity using clinical features, dosimetrics, and radiomics. The classification algorithms are LR, RF, and SVM. Acute toxicity was divided into four classes: *any toxicity*, *fatigue*, *esophagus toxicity*, and *lung toxicity*. Models were trained using all combinations of input features, classification algorithms, and outcome classes. The preprocessing of the data was a very large part of this project, resulting in a data set of 458 patients.

It is difficult to directly compare the AUC values found in this study to results from related work, because different data sets were used to evaluate the models. Moreover, four overarching acute toxicity outcome classes were predicted in this study, as opposed to predicting RP and other more specific types of toxicity in related work. Keeping this in mind, the best result found here, an AUC value of 0.86, is higher than any of the AUC values found in any of the related work, showing that our methods are very promising for future use. We will discuss the results found here and compare them more generally to related work below. Strengths and limitations of this study along with suggestions for future work will be discussed throughout this section.

In accordance with most related work, dosimetrics are among the best predicting features and clinical features can be a good addition to them in predicting toxicity. However, our results show that the addition of clinical features to dosimetrics does not increase the predictive performance very significantly, while Chen, Zhou, Zhang, et al. (2007), Chen, Zhou, Yin, et al. (2007a), and Chen, Zhou, Yin, et al. (2007b) show this improvement more clearly. This might be due to the difference in number of clinical and dosimetric features used in each study, as those three studies used 66 dosimetrics and 27 clinical features, while in our study, we used 201 dosimetrics per organ and only 9 clinical features. The addition of 9 clinical features to the dosimetrics will then likely result in a smaller increase in performance. The study by Krafft et al. (2018) was the only one that achieved poor results for clinical and dosimetric features with AUC scores around 0.5. Our study does bring forth very contrasting ideas to the related work regarding radiomics. Krafft et al. show that radiomics have great predictive power for RP and that there is no real difference between the different types of radiomics and Bousabarah et al. (2019) show that only first order radiomics are predictive of fibrosis. It might be that the radiomics are simply more predictive for RP and fibrosis than for (a broader range of) acute toxicities. Although RP is one of the acute toxicities considered in this study, it is only a very small part of all acute toxicities. Fibrosis is a toxicity that usually occurs three to six months after the last treatment, meaning that it would not fall under our definition of acute toxicity. Thus, it could be that radiomic features are more predictive for later occurring toxicities than for acute toxicities. Moreover, in our study, only shape and first order features were considered, while Krafft et al. and Bousabarah et al. also used other radiomic features. To get a better view of what type of radiomics are predictive of what specific outcome, more experiments should be performed using the different types of radiomic features that are available along with a wider range of outcomes to be predicted.

The main contribution of this study is that we have managed to create a data set that is very large compared to related studies. Compared to many machine learning studies, a data set of 458 instances is not a very large data set at all, but it is for medical studies such as this one. The largest data set found in related work consists of 235 patients (Das et al., 2008). This means that the data set created here is almost twice the size of the largest data set found in related work. This is the result of our interdisciplinary approach consisting of close consultation with radiation oncologists and very well thought out decision-making. The laborious process of preprocessing the data is described in detail, so that it can be reproduced in future work.

The way this project is set up allows for good comparison of the input types, classification algorithms, and outcome classes, because a model is trained and evaluated for every combination of input type, classification algorithm, and output class. It is also a very broad study, incorporating all stages of lung cancer, a wide range of patients, and many different types of inputs used in related work. This makes it more applicable to real life situations, where the variety of patients is enormous.

A limitation of this study is that dosimetrics and radiomics are only a simplification of the raw CT scan and dose distribution. Dosimetrics and radiomics are metrics calculated from the DVH and first order histogram, which are both simplifications of their respective raw image. A DVH reduces the three-dimensional dose distribution information into a two-dimensional graph. Then, the information is further simplified by taking single metrics (e.g. V_{20}, MLD) from this graph. The result of this is that spatial location information of the dose distribution is lost and thus not taken into account by the model when making a prediction. With radiomics, we have a somewhat similar situation. The radiomic features are calculated from the CT image, but in doing so, we do also neglect some subtle features. An improvement over the dosimetric features was suggested by Liang et al. (2019), who compared the predictive performance of dosimetric features and dosiomic features. As explained before, dosiomics are similar to radiomics, except they are taken from the dose distribution image instead of the CT image. They found that the models using dosiomics outperform models using dosimetrics, but they tested this on a data set of only 70 patients, hence we should be sceptical of this result. Another suggestion by the same authors was to use the raw CT scan and dose distribution images as input for a 3D CNN (Liang et al., 2020). This model outperformed their dosiomics model, but again, they trained it on the same data set of 70 patients, hence we should be sceptical of this result as well. Our suggestion for future work would be to use a 3D CNN inspired by Liang et al. on our data set of 458 patients. This 3D CNN will likely be able to capture features from the dose distribution and CT image better than the methods used for dosimetrics, radiomics, and even dosiomics. We are convinced that this is the next step in this line of research.

As discussed in sections 5.1 and 5.2, the dosimetrics and radiomics contained information of the separate lungs and bronchi instead of the left and right lung and bronchus combined. This likely resulted in a worse performance for the prediction of the *lung toxicity* class. To improve on this in future work, it would be a good idea to use the dosimetrics and radiomics of both lungs and both bronchi combined.

A shortcoming of this study that limits the usability of this method in practice is the exclusion of patients that have been treated more than once. In theory, the models proposed here can be used for subsequent treatments, but the outcome may be misleading, because the risk of developing acute toxicity is higher due to cumulative dose. Therefore, it would be useful to think of a good way to include non first-time treatments in the model.

Another interesting addition to the prediction of acute toxicity is to predict a specific grade of toxicity. That way, we cannot only predict whether acute toxicity will occur, but also how severe the toxicity will be. This might be helpful with creating a treatment plan. The patient and radiation oncologist can then in consultation decide what grade of toxicity is acceptable and adjust the treatment plan accordingly.

7 Conclusion

The data set created in this study is larger than any data set found in related work predicting radiation-induced toxicity in lung cancer patients. The preprocessing steps taken have been explained in detail, in order to be able to reproduce the process in future work. We found that, depending on the outcome class, it is possible to accurately predict acute toxicity. Especially for the *esophagus toxicity* class, we have seen that it is possible to predict acute toxicity well using the methods proposed here. Perhaps the most useful outcome class to predict in practice, the *any toxicity* class, can also be predicted reasonably well. Predicting any type of acute toxicity accurately allows the radiation oncologist to better create a treatment plan suited for a specific patient. *Fatigue* was more difficult to predict, but this is not very surprising considering the nature of this type of acute toxicity. *Lung toxicity* was not predicted very well, but we propose an improvement of the dosimetric and radiomic input features that might be able to solve this problem. It was found that dosimetrics are the most predictive features, closely followed by clinical features. The radiomics used in this study were found to be not predictive of acute toxicity at all. Our suggestion for future work is to use the raw CT scan and dose distribution images from our data set as input for a 3D CNN. We think that this is the next step in the prediction of acute toxicity.

References

Bishop, C. M. (2006). *Pattern recognition and machine learning.* Springer.

Bousabarah, K., Temming, S., Hoevels, M., Borggrefe, J., Baus, W. W., Ruess, D., … Treuer, H. (2019). Radiomic analysis of planning computed tomograms for predicting radiation-induced lung injury and outcome in lung cancer patients treated with robotic stereotactic body radiation therapy. *Strahlentherapie und Onkologie*, *195*(9), 830–842.

Bradley, A. P. (1997). The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern recognition*, *30*(7), 1145–1159.

Chen, S., Zhou, S., Yin, F.-F., Marks, L. B., & Das, S. K. (2007a). Investigation of the support vector machine algorithm to predict lung radiation-induced pneumonitis. *Medical physics*, *34*(10), 3808–3814.

Chen, S., Zhou, S., Yin, F.-F., Marks, L. B., & Das, S. K. (2007b). Using patient data similarities to predict radiation pneumonitis via a self-organizing map. *Physics in Medicine & Biology*, *53*(1), 203.

Chen, S., Zhou, S., Zhang, J., Yin, F.-F., Marks, L. B., & Das, S. K. (2007). A neural network model to predict lung radiation-induced pneumonitis. *Medical physics*, *34*(9), 3420–3427.

Cui, S., Luo, Y., Tseng, H.-H., Ten Haken, R. K., & El Naqa, I. (2019). Combining handcrafted features with latent variables in machine learning for prediction of radiation-induced lung damage. *Medical physics*, *46*(5), 2497–2511.

Das, S. K., Chen, S., Deasy, J. O., Zhou, S., Yin, F.-F., & Marks, L. B. (2008). Combining multiple models to generate consensus: Application to radiation-induced pneumonitis prediction. *Medical physics*, *35*(11), 5098–5109.

Das, S. K., Zhou, S., Zhang, J., Yin, F.-F., Dewhirst, M. W., & Marks, L. B. (2007). Predicting lung radiotherapy-induced pneumonitis using a model combining parametric lyman probit with nonparametric decision trees. *International Journal of Radiation Oncology* Biology* Physics*, *68*(4), 1212–1221.

De Ruyck, K., Sabbe, N., Oberije, C., Vandecasteele, K., Thas, O., De Ruysscher, D., … Thierens, H. (2011). Development of a multicomponent prediction model for acute esophagitis in lung cancer patients receiving chemoradiotherapy. *International Journal of Radiation Oncology* Biology* Physics*, *81*(2), 537–544.

Dowle, M., & Srinivasan, A. (2020). data.table: Extension of 'data.frame' [Computer software manual]. Retrieved from `https://CRAN.R-project.org/package=data.table` (R package version 1.13.0)

Fowler, J. F. (1989). The linear-quadratic formula and progress in fractionated radiotherapy. *The British journal of radiology*, *62*(740), 679–694.

Gayou, O., Das, S. K., Zhou, S.-M., Marks, L. B., Parda, D. S., & Miften, M. (2008). A genetic algorithm for variable selection in logistic regression analysis of radiotherapy treatment outcomes. *Medical physics*, *35*(12), 5426–5433.

Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., ... Oliphant, T. E. (2020). Array programming with NumPy. *Nature*, *585*, 357–362. doi: 10.1038/s41586-020-2649-2

Huang, E. X., Robinson, C. G., Molotievschi, A., Bradley, J. D., Deasy, J. O., & Oh, J. H. (2017). Independent test of a model to predict severe acute esophagitis. *Advances in radiation oncology*, *2*(1), 37–43.

Hunter, J. D. (2007). Matplotlib: A 2d graphics environment. *Computing in science & engineering*, *9*(3), 90–95.

Krafft, S. P., Rao, A., Stingo, F., Briere, T. M., Court, L. E., Liao, Z., & Martel, M. K. (2018). The utility of quantitative ct radiomics features for improved prediction of radiation pneumonitis. *Medical physics*, *45*(11), 5317–5324.

Krstajic, D., Buturovic, L. J., Leahy, D. E., & Thomas, S. (2014). Cross-validation pitfalls when selecting and assessing regression and classification models. *Journal of cheminformatics*, *6*(1), 1–15.

Lee, S., Ybarra, N., Jeyaseelan, K., Faria, S., Kopek, N., Brisebois, P., ... El Naqa, I. (2015). Bayesian network ensemble as a multivariate strategy to predict radiation pneumonitis risk. *Medical physics*, *42*(5), 2421–2430.

Liang, B., Tian, Y., Chen, X., Yan, H., Yan, L., Zhang, T., ... Dai, J. (2020). Prediction of radiation pneumonitis with dose distribution: A convolutional neural network (cnn) based model. *Frontiers in oncology*, *9*, 1500.

Liang, B., Yan, H., Tian, Y., Chen, X., Yan, L., Zhou, Z., ... Dai, J. (2019). Dosiomics: extracting 3d spatial features from dose distribution to predict incidence of radiation pneumonitis. *Frontiers in Oncology*, *9*, 269.

Luna, J. M., Chao, H.-H., Diffenderfer, E. S., Valdes, G., Chinniah, C., Ma, G., ... Simone II, C. B. (2019). Predicting radiation pneumonitis in locally advanced stage ii–iii non-small cell lung cancer using machine learning. *Radiotherapy and Oncology*, *133*, 106–112.

Munley, M., Lo, J., Sibley, G., Bentel, G., Anscher, M. S., & Marks, L. (1999). A neural network to predict symptomatic lung injury. *Physics in Medicine & Biology*, *44*(9), 2241.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... others (2011). Scikit-learn: Machine learning in python. *Journal of machine learning research*, *12*(Oct), 2825–2830.

R Core Team. (2020). R: A language and environment for statistical computing [Computer software manual]. Vienna, Austria. Retrieved from `https://www.R-project.org/`

Spencer, S. J., Almiron Bonnin, D., Deasy, J. O., Bradley, J. D., & El Naqa, I. (2009). Bioinformatics methods for learning radiation-induced lung inflammation from heterogeneous retrospective and prospective data. *BioMed Research International*, *2009*.

Su, M., Miften, M., Whiddon, C., Sun, X., Light, K., & Marks, L. (2005). An artificial neural network for predicting the incidence of radiation pneumonitis. *Medical physics*, *32*(2), 318–325.

Timmerman, R., Paulus, R., Galvin, J., Michalski, J., Straube, W., Bradley, J., ... Johnstone, D. (2010). Stereotactic body radiation therapy for inoperable early stage lung cancer. *Jama*, *303*(11), 1070–1076.

Valdes, G., Solberg, T. D., Heskel, M., Ungar, L., & Simone II, C. B. (2016). Using machine learning to predict radiation pneumonitis in patients with stage i non-small cell lung cancer treated with stereotactic body radiation therapy. *Physics in Medicine & Biology*, *61*(16), 6105.

Van Rossum, G., & Drake, F. L. (2009). *Python 3 reference manual.* Scotts Valley, CA: CreateSpace.

Wes McKinney. (2010). Data Structures for Statistical Computing in Python. In Stéfan van der Walt & Jarrod Millman (Eds.), *Proceedings of the 9th Python in Science Conference* (p. 56 - 61). doi: 10.25080/Majora-92bf1922-00a

Yu, H., Wu, H., Wang, W., Jolly, S., Jin, J.-Y., Hu, C., & Kong, F.-M. (2019). Machine learning to build and validate a model for radiation pneumonitis prediction in patients with non–small cell lung cancer. *Clinical Cancer Research*, *25*(14), 4343–4350.

Appendices

A PubMed Search Term

```
"Lung Neoplasms"[Mesh] OR "Lung cancer"[tiab] OR "Lung cancers"[tiab] OR "Lung
neoplasm"[tiab] OR "Lung neoplasms"[tiab] OR "Pulmonary neoplasm"[tiab] OR
"Pulmonary neoplasms"[tiab] OR "Pulmonary cancer"[tiab] OR "Pulmonary
cancers"[tiab] OR "Cancer of the lung"[tiab] OR "Cancer of lung"[tiab] OR "Lung
carcinoma"[tiab] OR "Lung nodule"[tiab] OR "Lung nodules"[tiab] OR "Lung
tumor"[tiab] OR "Lung tumors"[tiab] OR "Lung tumour"[tiab] OR "Lung
tumours"[tiab] OR "Small-cell carcinoma"[tiab] OR "SCLC"[tiab] OR "NSCLC"[tiab]
AND
"Radiotherapy"[Mesh] OR Radiation[tiab] OR radiotherap*[tiab] OR "X-ray"[tiab] OR
"SBRT"[tiab] OR "SABR"[tiab] OR Dose[tiab] OR Doses[tiab] OR Dosimetric[tiab] OR
Dosiomics[tiab]
AND
predict*[tiab] OR predictive value of tests[mh] OR score[tiab] OR scores[tiab] OR
scoring system[tiab] OR scoring systems[tiab] OR observ*[tiab] OR observer
variation[mh] OR estimat*[tiab]
AND
"Computing Methodologies"[Mesh] OR "Artificial Intelligence"[tiab] OR "AI"[tiab]
OR Learning[tiab] OR Machine[tiab] OR Machines[tiab] OR "CNN"[tiab] OR
"RNN"[tiab] OR "SVM"[tiab] OR "Neural Network"[tiab] OR "Neural Networks"[tiab]
OR Algorithm*[tiab] OR Bayesian[tiab] OR Markov[tiab] OR Dosiomics[tiab]
```

B CTC Registration Types

Acute kidney injury; Creatinine increased; Abdominal pain; Anorexia; Aspiration; Back pain; Brachial plexopathy; Cataract; Chest wall pain; Conjunctivitis; Constipation; Cough; Cystitis non-infective; Dehydration; Dermatitis radiation; Diarrhea; Dry mouth; Dysphagia; Dyspnea; Edema limbs; Enterocolitis; Erectile dysfunction; Esophageal perforati; Esophagitis; Fatigue; Fecal incontinence; Fibr deep connective tissu; Flatulence; Fracture; Gastrointestinal fistula; Gastrointestinal hemorrhag; Glaucoma; Hearing impaired; Hematuria; Hemorrhoids; Hoarseness; Hot flashes; Hypothyroidism; Insomnia; Intestinal stenosis; Laryngeal edema; Laryngeal mucositis; Laryngopharyngeal dysesthe; Lymphedema; Lymphedema breast/chest; Middle ear inflammation; Nausea; Nervous system disorders; Optic nerve disorder; Osteonecrosis of jaw; Pain; Pain breast; Pain in extremity; Pharyngeal hemorrhage; Pharyngeal stenosis; Platelet count decreased; Pneumonitis; Proctitis; Pulmonary fibrosis; Radiation recall reaction; Rectal fistula; Rectal hemorrhage; Rectal pain; Retinopathy; Salivary duct inflammation; Skin and subcutaneous tiss; Skin atrophy; Skin hyperpigmentation; Sore throat; Telangiectasia; Trismus; Urinary fistula; Urinary frequency; Urinary incontinence; Urinary retention; Urinary tract obstruction; Urinary tract pain; Urinary urgency; Vaginal discharge; Vaginal dryness; Vaginal hemorrhage; Vaginal inflammation; Vaginal stricture; Vomiting; Weight loss; White blood cell decreased; Dyspepsia