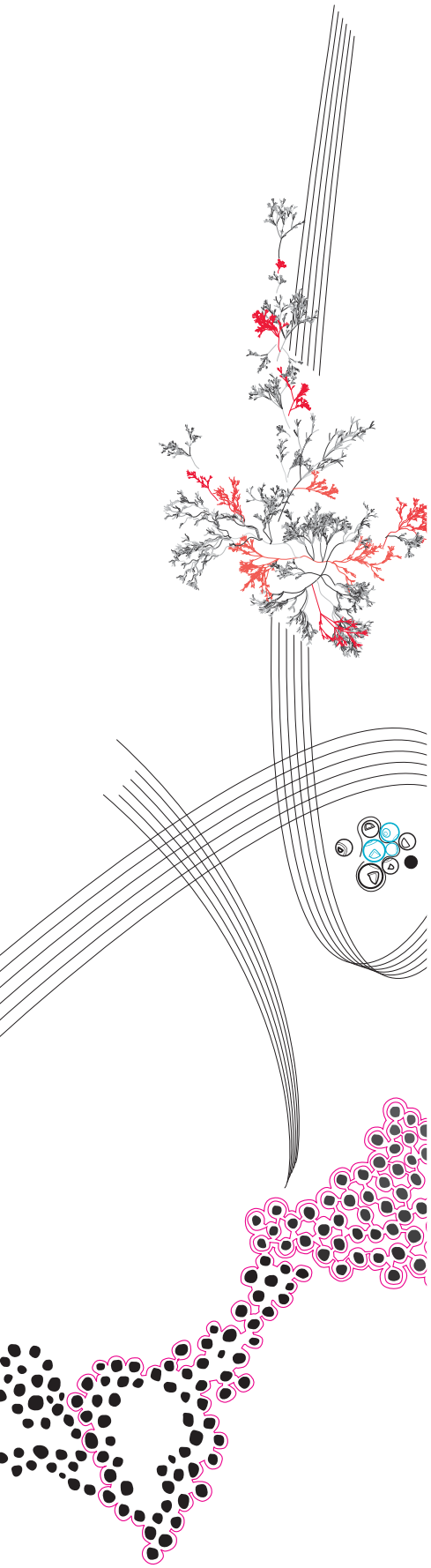


BSc Thesis Applied Mathematics

Prediction of treatment results for patients with rheumatoid arthritis

Aline de Jong

Supervisors: R. Boucherie, M. de Graaf, H. Moens

July, 2020

Department of Applied Mathematics
Faculty of Electrical Engineering,
Mathematics and Computer Science

UNIVERSITY OF TWENTE.

Prediction of treatment results for patients with rheumatoid arthritis

Aline M. de Jong

July, 2020

Abstract

The RAAD-score is a measurement for long term irreversible joint damage for patients with rheumatoid arthritis. The goal of this study is to predict the RAAD-score and to see which variables can explain the RAAD-score. Patient classification will be used for both. Multiple methods were considered for patient classification. The methods that are applied to the data are CART, random forest, and Naive Bayes. One of the importance measures from the random forest method is used to determine the most important variables. Based on the computations in this research it can be concluded that predicting the RAAD-score for patients with rheumatoid arthritis, using one of the mentioned methods, is not easy. Main reason for this is that the data set is complicated and on both numerical and classification aspects.

Keywords: Rheumatoid arthritis, RAAD-score, patient classification.

1 Introduction

Rheumatoid arthritis (RA) is a chronic joint inflammation, which if untreated results in irreversible joint damage. Medication may suppress inflammation and disease activity. The current guidelines advise a treat-to-target strategy. The treatment is aimed at lowering the disease activity, which is usually measured with the "disease activity score" (DAS). The outcome of treatment after five to ten years can be assessed and scored with the "Rheumatoid Arthritis Articular Damage"-score, or RAAD-score. This outcome may depend on baseline characteristics and interventions. Interventions, mostly treatment with various drugs, are determined by national and international guidelines. However, in routine medical care there is practice variation, while treatment strategies also change over time, with the development of new drugs. This study aims to investigate whether the RAAD-score can be explained by specific variables. The approach will be to compare mathematical data analysis methods, in particular CART, random forest, and Naive Bayes, for patient classification, prediction of RAAD-score, and effectiveness of applied treatments per patient. The following questions can be asked in order to answer the question whether the RAAD-score can be explained by specific variables:

1. On which attributes may the RAAD-score depend?
2. Which attributes explain the long-term outcome of RA as expressed by the RAAD-score?
   (a) Which data analysis methods are suitable for patient classification?
3. Is the RAAD-score in part explained by variation in the applied treatments?

1.1 Background

The RAAD-score of a patient is determined by physical examination of 35 small and large joints. Each joint is scored separately, 0 if no damage, 1 if partial irreversible damage, and 2 for severe irreversible damage [1]. That means that the range of the RAAD-score is in the integers from 0 to 70. For example, a score of 2 is applied when a knee has to be replaced by a joint prosthesis. A low RAAD-score, below 3, means there is little to no damage. A high RAAD-score, above 20, indicates that there are a lot of (severely) damaged joints.

2 Method

The classification of patients can be used to see which attributes determine the RAAD-score. The complete data set contains data of 1381 patients. Since to aim is to investigate the long-term outcome of patients with RA, not all data are used. More details about the data can be found in Section 3 of this article.

The programming language R is used for classification. There are multiple classification methods already available in R. The methods below are the methods that were considered.

Naive Bayes

The Naive Bayes algorithm is a simple method that uses Bayes' theorem for classification. Bayes' theorem states that

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}.$$

When using Naive Bayes for classification, the classification becomes the probability that an object belongs to a certain class, given the features of the object. In the formula above, A is the class, and B is the intersection of all the features of the object: $B_1 \cap B_2 \cap \cdots \cap B_n$, where n is the number of known attributes of the object. Since Naive Bayes assumes independence between all attributs and the denominator does not depend on the class [2],

$$P(A|B) \propto P(B_1 \cap B_2 \cap \cdots \cap B_n|A)P(A) = P(B_1|A)P(B_2|A)\dots P(B_n|A)P(A).$$

The likelihood that the class is A is then $P(B_1|A)P(B_2|A)\dots P(B_n|A)P(A)$. To get the probability that the class is A, the likelihood must be divided by the total likelihood across all possible classes.

Naive Bayes can only handle categorical data, so it is not ideal for data sets with many numerical attributes [2]. In order to apply Naive Bayes, all numerical attributes and the outcome, the RAAD-score, need to be categorised.

C5.0

The C5.0 algorithm makes decision trees, of which the leaf nodes correspond to the different classes. A decision tree starts with a root node, this is the first parent node. At each node a decision is made about the category or value of an attribute. After a decision is made, a new node, child node, is reached. In these child nodes, another decision is made based on an attribute and the child node becomes a parent node. This process continues until there is no further decision to be made and a leaf node is reached. Since, for C5.0, the leaf nodes represent the classes, the tree is called a classification tree.

C5.0 can handle both numerical AND categorical attributes. For the numerical attributes it makes binary splits and for categorical attributes two or more splits are used depending on the number of categories. To determine the attribute that should be used for the next

split/decision, the C5.0 algorithm uses information gain. The information gain is calculated using entropy. The entropy of a given segment of data S is the expected information content of S and is calculated as follows:

$$\text{Entropy}(S) = -\sum_k p_k \log_2 p_k,$$

where p_k is the proportion of objects that fall into class k. Then the information gain is obtained using

$$\text{Information Gain} = \text{Entropy}(S) - \sum_i w_i \text{ Entropy}(P_i),$$

where Entropy(S) is the entropy before the split, w_i is the proportion of objects in the i^{th} split, and Entropy(P_i) is the entropy in the remaining data after split i is made. The attribute with the highest information gain is used next for splitting.

Even though the attributes can be either categorical or numerical, the class variable needs to be categorical in order to apply C5.0. The leaf nodes should give the RAAD-score of a patient. Since the RAAD-score is numerical, the RAAD-score needs to be categorised.

CART

CART stands for Classification And Regression Tree. In the basis, CART is similar to C5.0. A decision tree created by CART also starts with a root node where a decision is to be made. After each decision, another node is reached. In the nodes, either a decision is made or the node is a leaf node if there is no further decision to make. With CART, each decision is a binary split. So for categorical attributes with more than two categories, the categories are grouped so that there are two groups that each go to a separate branch of the decision tree. For numerical attributes, a splitting point is determined.

Furthermore, the CART algorithm allows the outcome to be either numerical or categorical. If the outcome is categorical, it creates a classification tree. If the outcome is numerical, it creates a so called regression tree. However, a regression tree does not actually use regression to create a decision tree. Since the RAAD-score is numeric, a regression tree is obtained. Instead of having classes at the leaf nodes of the decision tree, the leaf nodes of the regression tree give the average value of the outcome of the patients in that leaf node.

The splitting criterion that is used depends on whether the tree is a classification or regression tree. When CART is used for building a classification tree, the splitting criterion that is used is the Gini index. The Gini index is calculated using the following formula:

$$\text{Gini} = 1 - \sum_k {p_k}^2,$$

where p_k is the proportion of objects that fall into class k. The Gini index is a value between 0 and 1. If the Gini index is 0, then all elements belong to the same class. If the Gini index is 1, then the objects are randomly distributed over the classes. The attribute with the smallest Gini index is used next for splitting when building a decision tree.

For building a regression tree, the goal of the CART algorithm is to find leaf nodes such that the decision tree minimises the sum of squared residuals (SSR). The SSR is given by:

$$\text{SSR} = \sum_{j=1}^{J} \sum_{i \in R_j} \left(y_i - \hat{y}_{R_j}\right)^2,$$

where R_j are the leaf nodes of the regression tree, y_i are the actual values of the objects in the leaf node R_j, and $\hat{y}_{R_j}$ is the mean of the outcome of the objects in the leaf node R_j.

Decision trees are very prone to over fitting. That means a tree will be very good at reproducing the training data, but may not perform as well for the testing data. In order to avoid over fitting, a technique called cost-complexity pruning can be used to remove some splits that do not improve the prediction. Now the goal is to minimise

$$\text{SSR} + (\text{size penalty}) = \sum_{j=1}^{J} \sum_{i \in R_j} \left(y_i - \hat{y}_{R_j}\right)^2 + \alpha J,$$

where SSR is as before, α is the complexity parameter, and J is the number of leaf nodes. A small value for α results in a very large tree, as the penalty for the tree size is very small. Taking a relatively large value for α results in a tree that only consist of a root node. The implementation of CART takes $\alpha = 0.01$ as default.

To find the best value for α, the approach is to start with a small value for α and then increase α in steps. At each step, the cross validation error is computed, the tree for which the α gives the smallest cross validation error is then used.

Random forest

Decision trees have high variance. So algorithms that create a single tree may give a very different decision tree for a slightly different subsets of the same data. Ensemble methods can be used to reduce the variance and increase the performance for predicting. One ensemble method is random forest. Random forest takes samples with replacement, bootstrap samples, from the training set and creates regression trees for each sample. The bootstrap samples each come with their own test set, the out-of-bag (OOB) observations. Each tree finds an estimate of the outcome of the object, $\hat{y}_1, \ldots, \hat{y}_B$, where B is the number of bootstrap samples. Then an overall prediction is made using the average of the found estimates.

Each time random forest makes a split, only a random subset m of the attributes is considered. For regression, $m = p/3$ is used, where p is the number of attributes. This way the trees have low correlation, since strong features are not used first in every tree and so the trees will differ from each other.

The downside of random forest, compared to a model with just one decision tree, is that the decision making process cannot be visualised.

M5'

M5' is an algorithm that builds a model tree. A model tree is very similar to a regression tree, the difference is in the leaf nodes. In the leaf nodes of a model tree, a linear regression model is used to determine the outcome. This is also why model tree generally gives better results than regression trees.

Since model trees use linear regression, all attributes and the outcome should be numerical. Therefore, the categorical data must be transformed to numerical.

Artificial Neural Networks

The method of artificial neural network is a so called black-box method. The transformation from input to output is obscured by an imaginary box. Artificial neural network models use the understanding of how a biological brain works to model the relations between input and output signals. Neural networks require all attributes to be numerical. This means the categorical attributes in the data set need to be transformed to numerical attributes.

Support Vector Machines

Support Vector Machines create a boundary between points in multidimensional space. The goal is to create a hyper plane that divides the space in to partition the data into groups of similar class values. Like Artificial Neural Network, Support Vector Machine is a black-box method. Furthermore, Support Vector Machines also require all attributes to be numerical. So the categorical attributes need to be transformed to numerical.

Due to time restrictions and the practical applicability of the methods to the complex data set, not all methods are applied. The CART algorithm, random forest, and Naive Bayes are the methods that were used to analyse the data.

3 Data

The data set contains the attribute values per patient. Each patient can be identified by an anonymous patient number. The attributes used for patient classification are presented in Table 3.1. Although there is global awareness of risk factors for future joint damage despite treatment, there is little information regarding the prediction of long-term damage. Each of the attributes in Table 3.1 may potentially influence the RAAD-score.

These attributes are not all independent from each other. For example, the ACR 2010 score depends on six other variables. One of those is BSE, which in part depends on age and BMI. Also, the variables about smoking are related to each other.

The original data base contains variables that represent the same attribute, e.g. "serology" reflects the presence or absence of rheumatoid factors (attribute RF) and/or anti-CCP antibodies (attribute CCP). In cooperation with a rheumatologist the data set was reduced to the presented set of attributes, which includes the most representative variables.

The data set contains both categorical and numerical valued attributes. The outcome, the RAAD-score, is numeric. Furthermore, there are some missing data. Most implementations of the methods can still be applied to the data despite the missing data. Only random forest does not allow missing values in the data.

Not all classification methods described in Section 2 can be applied directly to the data, for some methods the data need to be transformed.

3.1 Transformation from numerical to categorical

In the description of C5.0 and Naive Bayes, it is mentioned that some data need to be transformed from numerical to categorical. In order to do this, the numbers are put into categories, the so-called bins. Each bin is an interval, if a number is inside the interval, it will be put into that bin. In order to do this, appropriate cut points need to be determined.

3.2 Transformation from categorical to numerical

M5', Artificial Neural Networks, and Support Vector Machines are algorithms that require data to be numerical. For transforming categorical data to numerical data, there are two options. The first option is integer encoding. With integer encoding, every category of an attribute is assigned an integer value. The integer values have an ordered relationship with each other, while this might not be the case in the original categorical attribute(s). So this method is not suitable for categorical attributes that do not have an ordered relation. The second option is to use one hot encoding. Using this method does not result in numerical values that have an ordered relationship. This method creates new variable, so called dummy variables, based on each category of a categorical attribute. It will assign a 1 if an object belongs to the category, and a 0 if not.

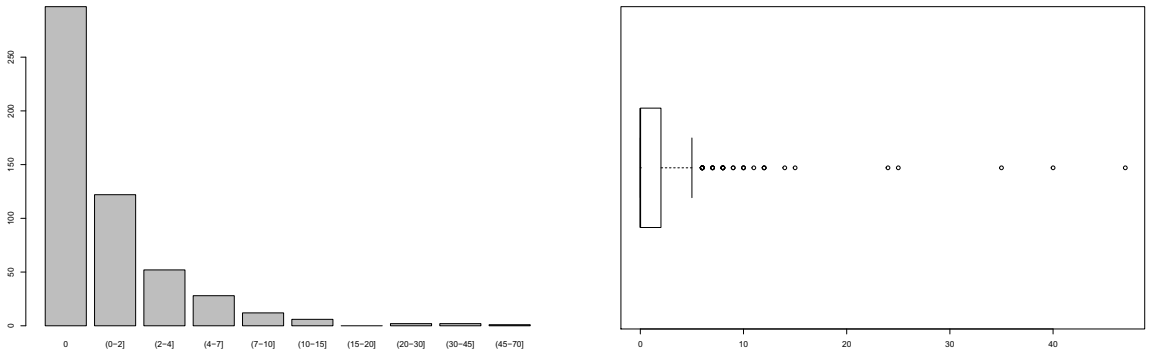
Table 3.1: Overview of attributes.

Attribute	Type	Description
Gender	cat.	Male or female.
Age at diagnosis	num.	Age of patient in years at the time of diagnosis.
BMI	num.	Body mass index.
Doctor	cat.	The doctor of the patient.
Affected joints	cat.	Number and size of affected joints.
Duration of arthritis	cat.	Whether the patient has symptoms for more or less than 6 weeks at the time of diagnosis.
Acute phase reaction	cat.	Whether BSE and/or CRP are normal or not.
BSE	num.	Measurement for the sedimentation rate of red blood cells (ESR).
CRP	num.	Measurement for the amount of the protein CRP in blood.
ACR 2010 score	num.	Score on scale 1-10 based on affected joints, serology, duration of arthritis, acute phase reaction, BSE, CRP.
Erosions	cat.	Whether there are erosions in joints before diagnosis.
Prednisolon	cat.	Whether the patient had prednisolon before diagnosis.
Smoke status	cat.	Whether the patient smokes or not, or has stopped smoking, at the time of diagnosis.
Type of tobacco	cat.	If applicable, type of tobacco the patient smokes/smoked.
Packyears	num.	Packyears = (years smoking)·(amount per day)/20, before and after diagnosis, until RAAD-score date.
Amount per day	num.	Number of cigarettes, cigars, etc. smoked per day.
RF	num.	Measure of auto-antibody in the blood common to RA.
CCP	num.	Measure of antibodies against the body's own protein CCP.
Steroids	cat.	Whether the patient got steroids and in which form.
Average DAS	num.	Average of the disease activity score.
Initial therapy	cat.	First treatment strategy that was used in a patient.
Start of b-DMD	num.	Time, in months, between diagnosis and the patient getting treated with b-DMD. B-DMD is an expensive drug that is work well.
Duration of b-DMD	num.	Duration of treatment with b-DMD in days.
Start of pred	num.	Time, in days, between diagnosis and the patient getting treated with prednisolon.
Duration of pred	num.	Duration of treatment with prednisolon in days.
Start of mtx	num.	Time, in days, between diagnosis and the patient getting treated with methotrexaat.
Duration of mtx	num.	Duration of treatment with methotrexaat, in days.
Start of bDMARD	num.	Time, in days, between diagnosis and the patient getting a bDMARD type of treatment. bDMARD are biological Disease Modifying Anti-Rheumatic Drugs, which are relatively expensive.
Duration of bDMARD	num.	Duration of treatment with bDMARD, in days.
Start of cDMARD	num.	Time, in days, between diagnosis and the patient getting a cDMARD type of treatment. cDMARD are classical Disease Modifying Anti-Rheumatic Drugs.
Duration of cDMARD	num.	Duration of treatment with cDMARD, in days.

4 Results

Since RA is a progressive disease, the duration of RA plays a big role in the height of the RAAD-score. Therefore only data of patients with a diagnosis in 2004 or later, and with a disease duration of more than four years are used for analysis. This leaves a data set containing 522 patients. Of these patients 90% of the data are used for training and the other 10% are used for testing.

Figure 4.1 displays the distribution of the RAAD-score using both histograms and box plots. In the histogram, it can be seen that most patients have a RAAD-score of 0, and the amount of patients decreases in each category of the RAAD-score as the RAAD-score increases. The horizontal box plot also shows that most patients have a RAAD-score of 2 or less, the dots in the box plot show potential outliers.



(A) Histogram. (B) Box plot.

FIGURE 4.1: Distribution of RAAD-scores.

4.1 CART

The CART algorithm is the only method that can be applied directly to the data without transformation, so this method will be investigated first. Applying the CART algorithm to the training data results in the decision tree as shown in Figure 4.2.

The tree is read from top to bottom. The first split is based on the BMI of the patient. If this is lower than 19.57, then node 9 is reached, it can be seen that there are some patients with a low BMI that have a high RAAD-score. The box plot in this node shows that the RAAD-score is wide spread for these patients as the median is quite low, while the mean RAAD-score of patients in this node is 12.9. This difference can be explained by the fact that there are only 11 patients in this node with wide spread RAAD-scores.

If the BMI of the patient is higher than or equal to 19.57, then there is another decision to be made in the next node, node 2. The split in node 2 is based on BSE. The tree can be read continuing this way until the patients have reached a leaf node. The box plots in the nodes do not show what the average RAAD-scores of the patients in the nodes are.

From the decision tree, the following rules are used for determining the RAAD-scores of patients. The first number is the RAAD-score, followed by when this score is assigned.

- 12.91 when BMI $<$ 19.57,
- 8.57 when BMI $\geq$ 19.57 & BSE $\geq$ 87.5,
- 0.95 when BMI $\geq$ 19.57 & BSE $<$ 87.5 & Lft.bij.start $<$ 63.5,
- 1.98 when BMI $\geq$ 19.57 & BSE $<$ 87.5 & Lft.bij.start $\geq$ 63.5.

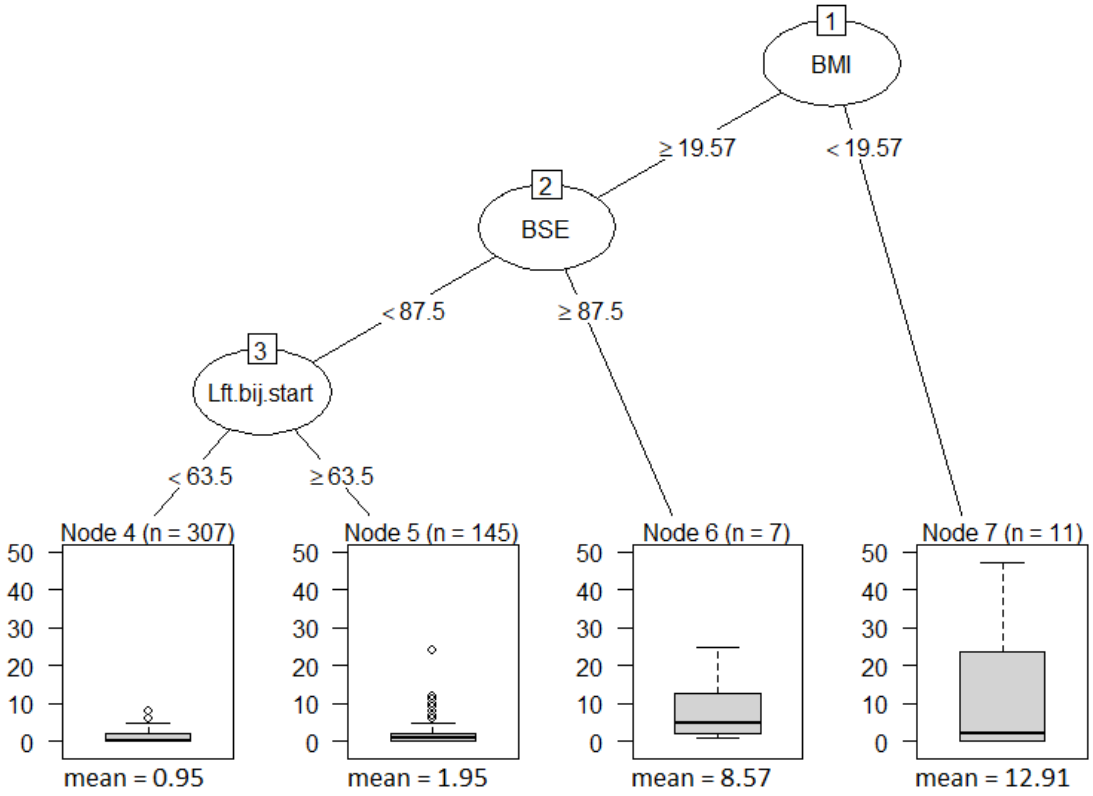


FIGURE 4.2: CART decision tree.

Next to the decision tree and the corresponding rules, the algorithm also gives an indication of the variable importance with respect to the other attributes, see Table 4.1. The variable importance is based on the reduction of SSR per attribute, and scaled so that the total sums up to 100.

TABLE 4.1: Variable importance of regression tree.

BMI	BSE	Age at start	Start of pred.
74	19	6	1

4.1.1 Model performance

Using the decision tree and the testing data set, it can be predicted how the model performs. The summary statistics of the predicted and the actual RAAD-scores of the test set can be found in Table 4.2. The summary statistics show the minimum, 25% quantile (1st. Qu.), 50% quantile (median), 75% (3rd Qu.) quantile, maximum, and the mean. The quantiles, together with the minimum and maximum say something about the distribution to RAAD-scores. The mean gives the average RAAD-score of the patients.

Since the algorithm takes the average RAAD-score of the patients in the training set in each node, it is expected that the lowest RAAD-scores of the prediction are not equal to zero. Overall, the decision tree model does seem to do quite well at predicting the RAAD-score. However, the prediction of a patient having a RAAD-score of 12.91 might not be the same patient that has an actual RAAD-score of 14. Therefore, other measures for model performance need to be examined.

TABLE 4.2: Summary statistics of the predictions using the decision tree and the testing data.

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
actual	0.00	0.00	0.00	2.00	2.25	14.00
predicted	0.95	0.95	0.95	1.44	1.98	12.91

Another indicator of model performance that can be used, is correlation. The correlation measures the relation between two vectors, it gives a value between -1 and 1. A correlation close to -1 or 1 indicates a strong linear relation. A correlation around 0 indicates there is most likely no relation between the two vectors. The correlation between the predicted and the actual RAAD-scores of the test set is 0.35. This does not indicate a very strong relation between the predicted and actual RAAD-scores.

Furthermore, the mean squared error (MSE) and the mean absolute error (MAE) can be used as a measure for model performance. The MSE is related to the SSR, the aim of the regression tree is to minimise the SSR as described in Section 2. The MAE gives insight in the difference between the prediction and the actual RAAD-score on average. Let n be the number of patients in the test set, then the MSE is given by:

$$\text{MSE} = \frac{\text{SSR}}{n} = \frac{1}{n}\left(\sum_{j=1}^{J}\sum_{i\in R_j}\left(y_i - \hat{y}_{R_j}\right)^2\right),$$

and the MAE is given by:

$$\text{MAE} = \frac{1}{n}\sum_{j=1}^{J}\sum_{i\in R_j}\left|y_i - \hat{y}_{R_j}\right|.$$

For both measures hold that the closet the value to zero the better the model performs. The MSE is mostly used to compare performance of methods. The MSE, using the predicted and actual test data, is 9.26. The MAE is found to be 1.93. This indicates that the model's prediction differs on average 1.93 form the actual RAAD-score.

4.2 Pruning CART

Before the pruning can be done, a very large regression tree has to be created. As described in Section 2, this can be done by setting the complexity parameter α very small at 0.00001. The tree obtained with $\alpha = 0.00001$ has 40 splits, where the tree before had 3 splits.

Now, analysing the cost parameter values and the corresponding cross-validation errors, the smallest cross-validation error is when $\alpha = 0.046$. The full table containing the different cost parameters and the corresponding cross-validation errors can be found in Appendix A. This α gives a tree with only 1 split and is the same split as the first split in Figure 4.2. The pruned tree can be found in Figure 4.3, and has the following rules:

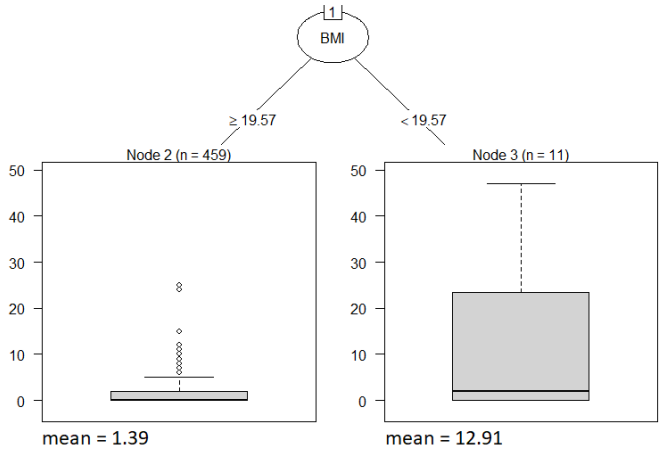


FIGURE 4.3: Pruned CART decision tree.

- 12.91 when BMI < 19.57,
- 1.39 when BMI ≥ 19.57.

The only variable that is important in the pruned tree is BMI.

In order to compare the performance of the pruned regression tree to the regression tree of Section 4.1, the same measures are used. The summary statistics of the pruned model are displayed in Table 4.3. From the summary statistics, it can be seen that the model performs worse for low RAAD-scores, but the predicted mean comes closer to the actual mean than before. The fact that the prediction for the maximum RAAD-score is the same as before can be explained since the original and the pruned tree have a corresponding leaf node.

TABLE 4.3: Summary statistics of the predictions using the pruned decision tree and the testing data.

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
actual	0.00	0.00	0.00	2.00	2.25	14.00
predicted	1.39	1.39	1.39	1.61	1.39	12.91

The correlation between the prediction from the pruned tree and the actual testing data is 0.36, which is slightly better than with the original CART model.

The MSE of the pruned tree is 8.96, where it originally was 9.26. This would indicate that the pruned model performs better than the original. However, the MAE of the pruned model is 2.05, where it originally was 1.93, and indicates that the original model performs better.

Taking all performance measures into account, it cannot be concluded whether the original or the pruned decision tree performs better.

4.3 Random forest

Another method that uses regression trees is random forest. It creates a lot of trees and takes the average of all the outcomes of the separate trees. Unfortunately, random forest is not able to handle missing data. Therefore the missing data need to be removed from the data set. Some attributes are missing for more than 50 patients. The same training and testing sets are used as before. In order to still have a reasonable size data set, these attributes are not used for the random forest. The removed attributes are: type of tobacco, packyears, amount per day, and CCP. For the remaining data, the patients that have missing values for one or more attributes are removed from the data. This results in a training set of 383 patients, and a test set of 39 patients.

In order to determine the number of trees for the random forest, first a random forest with a lot of trees, 1000 in this case, is created. The MSE of each size random forest is computed, the size with the smallest MSE is used for the random forest. The number of trees that is used for the random forest is 127.

The random forest has two measures for variable importance. The first is %IncMSE. This is the increase in MSE of predictions, estimated with the OOB observations, as a result of a variable being permuted. First the MSE of the full random forest is calculated with the OOB observations, let this be MSE_0. Then for each variable j in the model, the variable is permuted (randomly the values are shuffled). A new model is created with the permuted variable j and again the MSE error is calculated using the OOB observations, call this MSE_j. Now the %IncMSE of variable j is $\text{MSE}_0 - \text{MSE}_j$ averaged over all trees in the random forest and normalised by the standard deviation of the differences.

The second variable importance measure is IncNodePurity. It is the total decrease in SSR from splitting on the variable, averaged over all trees. For both measures of variable importance, a higher score indicates higher variable importance. The variable importance of the random forest, using both measures, is given in Figure 4.4, a table with the exact values of the importance measures can be found in Appendix B.

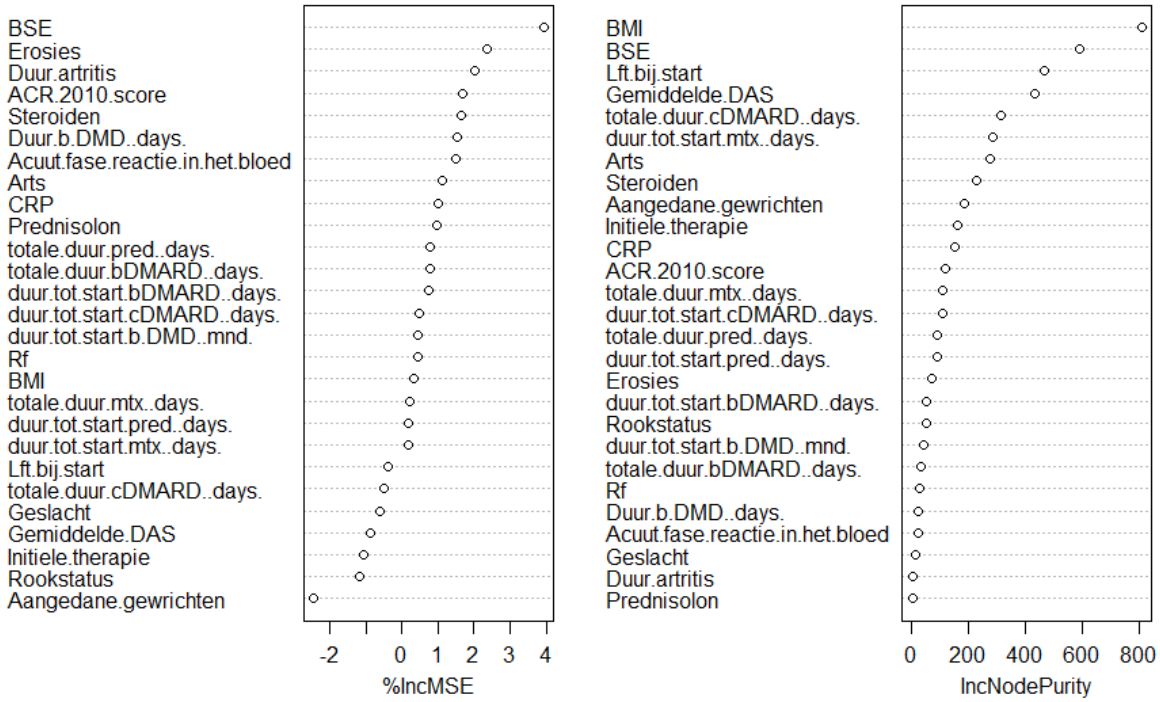


FIGURE 4.4: Importance of variables in random forest.

Both measures indicate that BSE is an important variable that can explain the RAAD-score. However, the IncNodePurity is more unstable and bias, since it may vary each model run and it favours variables with many levels. Therefore the %IncMSE is used to determine the important variables.

In order to see how an attribute influences the prediction of the RAAD-score, one should look at all separate decision trees to say something about this.

4.3.1 Model performance

In order to compare the performance of the random forest model to the original and the pruned regression tree, the same measures are used. The summary statistics of the prediction and the actual RAAD-scores of the test set can be found in Table 4.4. The summary statistics imply that the random forest model predicts the outcome for patients with a low RAAD-score quite close. For patients with a higher RAAD-score the random forest seems not to predict the outcome very well.

TABLE 4.4: Summary statistics of the predictions using random forest and the testing data.

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
actual	0.00	0.00	1.00	2.28	3.50	14.00
predicted	0.21	0.81	1.27	1.59	1.96	6.43

The correlation between the prediction of the random forest and the actual test data is 0.51. This is higher than previously found when using the CART algorithm, and indicates a moderate relation. The MSE of this model is 8.76, and the MAE is 1.87. These are both lower than the MSEs and MAEs seen before in the CART model.

However, since the training and testing data are not the same as before, the comparison is not fair. In order to do a fair comparison, the CART algorithms (original and pruned) need to be applied to the new data. The summary statistics in Table 4.5 and the performance measures in Table 4.6 show the comparison between the algorithms.

The performance measures in Table 4.6 show that the random forest does perform better than both the original and pruned CART models.

Table 4.5: Summary statistics of the predictions and the testing data.

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
actual	0.00	0.00	1.00	2.28	3.50	14.00
predicted with CART	1.10	1.10	1.10	1.54	1.10	9.50
predicted with pruned CART	1.48	1.48	1.48	1.68	1.48	9.50
predicted with random forest	0.21	0.81	1.27	1.59	1.96	6.43

Table 4.6: Performance measures of different models.

	correlation	MSE	MAE
CART	0.31	10.65	2.14
pruned CART	0.38	9.82	2.08
random forest	0.51	8.76	1.87

4.4 Naive Bayes

Where CART and random forest give an indication of which variables are important for predicting the RAAD-score, Naive Bayes only does classification. As stated in Section 2, all attributes and the outcome need to be categorical in order to apply Naive Bayes.

For Naive Bayes, the attributes that have %IncMSE above 1 are used for classification with Naive Bayes. These attributes are: BSE, erosions, duration of arthritis, ACR 2010 score, steroids, duration of b-DMD, acute phase reaction, and doctor. The variables BSE, ACR 2010 score, and duration of b-DMD need to be transformed from numerical to categorical. The other attributes are already categorical. Table 4.7 shows the intervals for the categories of each attribute. The cut points have been chosen such that each category contains approximately the same number of patients. For the duration of b-DMD, this was not possible since most patients never started this treatment. The patients that never started with b-DMD are put in a separate category, the other patients are divided over four intervals such that each interval contains approximately the same number of patients.

Table 4.7: Attributes from numerical to categorical and their cut points.

Attribute	Categories				
BSE	≤ 11	$(11, 19]$	$(19, 29]$	$(29, 43]$	> 43
ACR 2010 score	≤ 5	$(5, 6]$	$(6, 7]$	$(7, 8]$	> 8
duration of b-DMD	0	$(0, 1139]$	$(1139, 1990]$	$(1990, 2740]$	> 2740

The RAAD-score is categorised using the same intervals as in Figure 4.1a: 0, $(0, 2]$, $(2, 4]$, $(4, 7]$, $(7, 10]$, $(10, 15]$, $(15, 20]$, $(20, 30]$, $(30, 45]$, and $(45, 70]$. These categories are selected in consultation of a rheumatologist.

The Naive Bayes algorithm is trained with the training data, which contain 90% of the patients. The other 10% is used to compare the actual categories of the RAAD-sore to the predicted categories. This comparison can be found in Table 4.8. Since Naive Bayes has categorical output, as opposed to numerical, the performance measures used before cannot be applied here.

TABLE 4.8: Cross table of Naive Bayes predictions.

	Actual					
Predicted	0	$(0, 2]$	$(2, 4]$	$(4, 7]$	$(10, 15]$	Row total
0	27	9	4	1	0	40
$(0, 2]$	2	2	0	0	1	5
$(2, 4]$	0	1	1	0	1	3
$(4, 7]$	2	1	0	0	0	3
Column total	31	13	5	1	2	52

It can be seen that patients are quite often predicted to have a RAAD-score of 0, where this in not actually the case. From the table, the accuracy can be calculated:

$$\text{Accuracy} = \frac{\text{number of correctly predicted outcome}}{\text{total patients in test set}} = \frac{27 + 2 + 1}{52} \approx 0.58.$$

An accuracy of 58% is not very high. The model classifies quite a lot of patients to have a RAAD-score of 0, while the actual RAAD-score is higher. Patients with an actual RAAD-score of 0 get correctly classified. But the model also classifies lots of other patients at 0, while their actual RAAD-score is higher.

There are multiple explanations for the model not to perform very well. First, numerical attributes needed to be transformed to categorical before Naive Bayes could be applied. This does not contribute to the accuracy of the model. Second, more than half of the patients have a RAAD-score of 0, as can be seen in Figure 4.1a. These patients might all have very different attribute values, which might lead to the model favouring the class with a RAAD-score of 0.

Furthermore, Naive Bayes assumes that all attributes are independent from each other. With this data, this is not the case. For example, the ACR 2010 score depends, among other variables, on duration of arthritis, acute phase reaction, and BSE. Also the initial therapy may very well be related to the doctor of the patient.

5 Conclusion

This research was performed in order to discover which mathematical methods are suitable for for predicting the RAAD-score of patients with rheumatoid arthritis, and to find whether this score can be explained by certain variables. Of the available methods, three methods were applied to the data: CART, random forest, and Naive Bayes.

The classification of patients, and the prediction of their RAAD-score, proved to be complicated. There are lots of variables that may have an effect on the outcome, and the predictions do not seem to be very accurate.

The CART algorithm and the random forest gave insight in which attributes are important when predicting the RAAD-score. The %IncMSE measure of the random forest is the most robust and is used to decide which attributes can explain the outcome. These attributes are:

BSE, erosions, duration of arthritis, ACR 2010 score, steroids, duration of b-DMD, acute phase reaction, and doctor. These are also the variables that where used to train the Naive Bayes model. The Naive Bayes did not perform well with an accuracy of only 58%. This could be in part be explained by the complicated data, the transformation of data, and the assumed independence between variables.

Of the three methods that were applied to the data, random forest performed best when it comes to predicting the RAAD-score of patients. Next to this, random forest also gives a robust measure for variable importance. Based on this study, random forest is suggested as a mathematical method to both predict and explain the RAAD-score.

However, it might be worthwhile in future research to explore other methods that could perform better when it comes to predicting, and maybe explaining, the RAAD-score. Also, it might be an option to apply the used models to a bigger data set as more data becomes available, as it is known that the amount of available data influences the accuracy of the predictions.

Acknowledgements

I want to thank my supervisors R. Boucherie, M. de Graaf, and H. Moens for helping me in understanding the R programming language, and supplying and understanding the data of patients with RA. Also, I want to thank the people close to me for their support.

References

[1] T. R. Zijlstra, H. J. Bernelot Moens, and M. A. Bukhari, "The rheumatoid arthritis articular damage score: First steps in developing a clinical index of long term damage in RA," *Annals of the Rheumatic Diseases*, vol. 61, pp. 20–23, 1 2002.

[2] B. Lantz, *Machine Learning with R*. Birmingham: Packt Publishing Ltd., second ed., 2015.

A Cost complexity parameter for CART decision tree

In the table, the cross-validation error is denoted as 'xerror'.

TABLE A.1: Cost complexity parameter and corresponding cross-validation error.

Complexity parameter α	n split	rel error	xerror	xstd
0.178099986	0	1.00000	1.00202	0.35698
0.045805216	1	0.82190	0.96699	0.26698
0.013090727	2	0.77609	0.99227	0.26784
0.008967565	3	0.76300	1.01958	0.26733
0.007314297	5	0.74507	1.02030	0.26548
0.006383011	6	0.73775	1.01646	0.26673
0.005324613	7	0.73137	1.03018	0.26820
0.005249864	9	0.72072	1.02880	0.26823
0.004327849	11	0.71022	1.03397	0.26839
0.004187611	12	0.70589	1.03619	0.26838
0.003542870	13	0.70171	1.03189	0.26714
0.002786289	14	0.69816	1.02985	0.26713
0.002685929	17	0.68981	1.03223	0.26712
0.002590203	18	0.68712	1.03385	0.26711
0.002299863	19	0.68453	1.03141	0.26694
0.002271724	20	0.68223	1.03141	0.26694
0.002206665	21	0.67996	1.03090	0.26694
0.001859298	22	0.67775	1.02756	0.26801
0.001632541	23	0.67589	1.02927	0.26801
0.001534011	24	0.67426	1.03045	0.26800
0.001004791	25	0.67273	1.03222	0.26800
0.000830883	26	0.67172	1.03210	0.26800
0.000696784	28	0.67006	1.03585	0.26871
0.000567865	29	0.66936	1.03504	0.26871
0.000482623	30	0.66879	1.03518	0.26871
0.000395735	31	0.66831	1.03619	0.26937
0.000299526	32	0.66792	1.03673	0.26937
0.000279807	33	0.66762	1.03782	0.26937
0.000255099	34	0.66734	1.03791	0.26937
0.000251068	35	0.66708	1.03791	0.26937
0.000214267	36	0.66683	1.03791	0.26937
0.000199145	37	0.66662	1.03857	0.26938
0.000056019	38	0.66642	1.03937	0.26998
0.000015657	39	0.66636	1.03950	0.26998
0.000010000	40	0.66635	1.03916	0.26998

B Variable importance random forest

Table B.1: Variable importance of random forest.

Attribute	%IncMSE	IncNodePurity
Geslacht	-0.6078274	11.886931
Arts	1.1107536	276.937520
Aangedane.gewrichten	-2.4541231	183.355937
Duur.artritis	2.0208749	3.460980
Acuut.fase.reactie.in.het.bloed	1.4770729	21.566630
BSE	3.9569586	591.195417
CRP	0.9890549	153.127449
ACR.2010.score	1.6938300	119.456026
Erosies	2.3664290	70.452218
Prednisolon	0.9775856	2.949803
Rookstatus	-1.1923082	50.212746
Rf	0.4328144	27.812808
Steroiden	1.6314448	228.977837
BMI	0.3153830	813.391498
Lft.bij.start	-0.3925925	467.008760
duur.tot.start.b.DMD..mnd.	0.4566311	40.534761
Duur.b.DMD..days.	1.5293312	24.755168
Gemiddelde.DAS	-0.8870447	432.827346
Initiele.therapie	-1.0772182	161.216640
duur.tot.start.pred..days.	0.1879447	89.078654
totale.duur.pred..days.	0.7667765	90.853192
duur.tot.start.mtx..days.	0.1581210	283.881614
totale.duur.mtx..days.	0.2268293	110.980177
duur.tot.start.bDMARD..days.	0.7494410	52.079925
totale.duur.bDMARD..days.	0.7609155	34.075975
duur.tot.start.cDMARD..days.	0.4727015	109.033823
totale.duur.cDMARD..days.	-0.5141646	313.108337