04250 - Introduction to Machine Learning and Data Mining

Project 2 - Supervised Learning, E17

April 4, 2017

This report is written in collaboration by the following students:

Christian Kirstein Thygesen, s123871

Henriette Steenhoff, s134869

DTU Compute
Institut for Matematik og Computer Science

Contents

1 Regression Christian Kirstein Thygesen **1**

1.1 Linear regression with forward selection 1

1.2 Prediction of a new data observation 2

1.3 Fitting an Artificial Neural Network 3

1.4 Statistical evaluation of performance difference 3

2 Classification Henriette Steenhoff **5**

2.1 Description of classification problem 5

2.2 K-Nearest Neighbours (KNN) 6

2.3 Decision Tree 7

2.4 Logistic regression 9

2.5 Statistical comparison of performance 10

1 Regression

Christian Kirstein Thygesen

The South African Heart Disease dataset presents several interesting options for regression problems. In this report we have chosen to focus on predicting the risk of a patient suffering from coronary heart disease, by looking at the other nine attributes. However, in this part we will try to apply regression models, and thus, it makes no sense to try and predict a binary value such as the CHD attribute. Therefore, we will focus on predicting the variable SBP - Systolic Blood Pressure in this section.

1.1 Linear regression with forward selection

Forward selection is an algorithm that starts out with an empty model, and sequentially adds features to the model, testing whether the addition of a new feature improves the performance of the model. Thus, by this approach it is possible to extract a subset of features, that are sufficient to perform as good as or better than the full model.

Linear regression with forward selection was performed with at two-layer cross-validation, with a five-fold outer cross validation loop and a 10-fold inner cross validation loop. This means, that forward selection regression is performed on 5 different partitions of the dataset, and each forward selection process is evaluated 10 times to estimate the optimal forward selection process.

In each cross validation fold, the model chose some of the same attributes, with age being present in all folds, and adiposity in all but one (Fig 1).

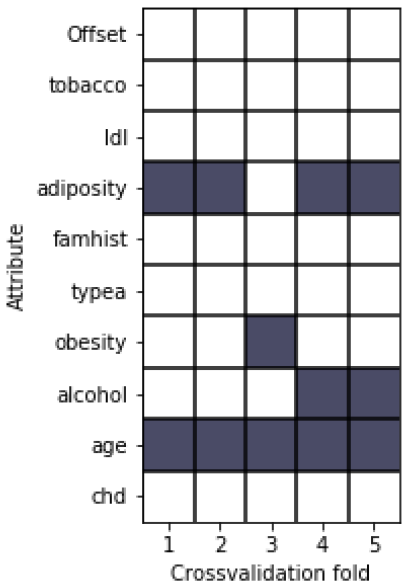


Figure 1: Subset features in each cross validation fold.

Regression model metrics		
Metric	Without feature selection	With feature selection
Training error	340.062	343.224
Test error	356.931	355.888
R^2 train	0.188	0.181
R^2 test	0.136	0.139

Table 1: Metrics of linear regression with and without forward feature selection.

Looking at the output of the model, we see that linear regression with forward selection does not perform better than a model without feeature selection (table 1). All the metrics from table X (training error, test error, R^2 train, and R^2 test), are almost identical suggesting, that this approach does not contribute significantly the regression model. The training and test error are both pretty high, which could suggest that this model is not ideal for predicting the systolic blood pressure of a patient.

Looking in to the possibility of transforming features by looking at the residual error for each of the attributes chosen in the cross validation folds, did not provide a solution to the poor regression model, as there were no clear observable trends in the residual error (figure 2).

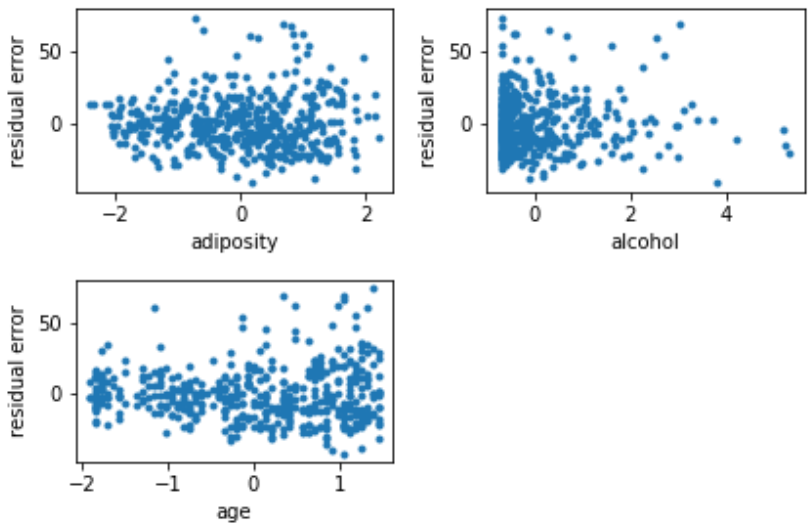


Figure 2: Residual error of features selected in cross validation loop 5.

1.2 Prediction of a new data observation

Given this prediction model, if we look deeper into the model from cross validation fold 5, we see that the model has chosen to use only three attributes; adiposity, alcohol and age (figure 1). As a result, if you wished to predict the systolic blood pressure of a patient, basically, you would need only those three attributes. The model then weights these three attributes given the estimated coefficients in

table 2. Age seems to have the largest impact on the prediction (an estimated coefficient of 5.453), followed by adiposity (3.691), and alcohol (1.947). Gathering this information we can represent the linear regression model from the general equation (equation 1)

$$f(x, w) = w_0 + w_1 * x_1 + w_2 * x_2 + w_3 * x_3 \tag{1}$$

to equation 2.

$$y = 138.327 + 3.691 * x_1 + 1.947 * x_2 + 5.453 * x_3 \tag{2}$$

Feature	Estimated coefficient
Adiposity	3.691
Alcohol	1.947
Age	5.453
(Intercept)	(138.327)

Table 2: Estimated coefficients of the selected model from forward feature selection. Model was generated from standardized data.

1.3 Fitting an Artificial Neural Network

An artificial neural network (ANN) can also be used for regression, and an ANN was fitted to the standardized data using 5-fold cross validation, 1 hidden layer, 50 hidden neurons, a learning goal of $1 * 10^{-10}$, and maximum number of epochs of 1000. The neural network was able to fit pretty well to the data with a mean square error of 30.788. Figure 3 shows that the ANN was able to fit almost perfectly, as the estimated values aligned perfectly with the actual values.

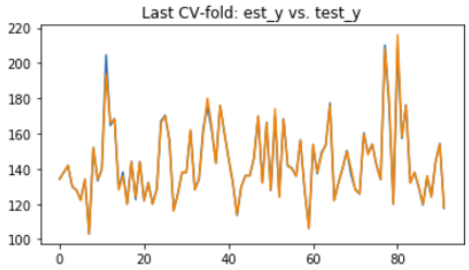


Figure 3: Estimated y values vs actual test y values.

1.4 Statistical evaluation of performance difference

The performance of this neural network was then compared to the performance of the linear regression model with forward selection, in order to see if any of the models performed significantly better than the other, when using the mean squared error as the performance measure.

This process was carried out by doing a 5-fold cross validation, where in each fold, a linear regression model with forward selection was generated with an internal cross validation level of 10, as well as a neural network with the same parameters as mentioned previously. The mean squared error was calculated in each cross validation fold and compared by a two-sided t-test with $\alpha = 0.05$.

As it turned out, the linear regression model looked to have a better performance, figure 4, and the t-test revealed that they were in fact significantly different, indicating, that the best model for this problem would be the linear regression model. Both models, however, performed significantly better compared to simply predicting the output to be the average of the training data output, with a huge improvement on the mean squared error.

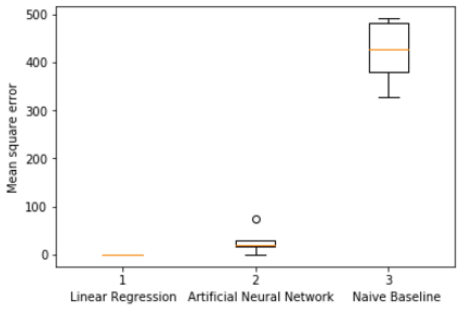


Figure 4: Comparison of the mean squared error of models of linear regression, artificial neural network, and naive base baseline.

2 Classification

Henriette Steenhoff

2.1 Description of classification problem

Since the different observations were hard to analyse with few features, as seen in Project 1, the methods of supervised learning will now be used in order to try and predict whether a male in the dataset has coronary heart disease or not.

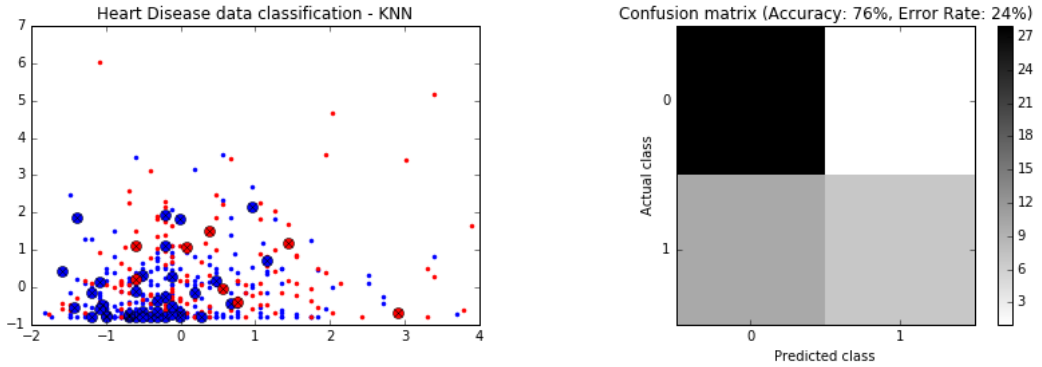
The group has chosen 3 different classification models: K Nearest Neighbour (KNN), Decision Trees and Logistic Regression. Using KKN might not be prefereable since this method tries to classify the different observations based on its nearest neigbours (which was really messy in Project 1). Decision Trees is thought to perform better as it weighs the different variables and tries to make the most optimal split when separating the samples.

For all methods, the same split (into training and test set) has been evaluated using two-layer cross-validation. The split is a $20/80$ split, where the training is performed on 80% of the data and the test on the remaining 20%. The cross-validation is 10-fold and for the decision tree, a maximum tree depth was set to 20, for KNN the numbers of neighbours to 50. From initial analysis of the decision tree, it was decided to use `gini` as impurity measure since this gave a better misclassification rate, as will be discussed later.

The following sections will go through the different models one by one. Some basic information about the different methods will be provided and the reason for choosing a specific model will be discussed. Subsequently, the performance results will be compared for each model and how to classify a new observations for each model will be explained – here the focus will be on the model with the optimal setting of the parameters as estimated by the two-layer cross-validation.

2.2 K-Nearest Neighbours (KNN)

Below is the confusion matrix and a scatter plot of the KNN training and test data. We tested on 30 neighbours and used the 2-norm, Euclidian distance as is the default value.



(a) (b)

Figure 5: Figure a) Scatterplot colored as classified by the KNN model. Figure b) the confusion matrix associated with the KNN model giving accuracy and error rate for the model.

Looking at the confusion matrix, we can easily calculate the accuracy (how often the classifier is right) and the error rate (how often the classifier is wrong) with the following equations:

$$Accuracy = \frac{11 + 10}{N}, Errorrate = \frac{01 + 00}{N},$$

where the ones indicates true and the zeros false.

This is what is indicated in the title of the right figure. The model classifies correctly 76% of the time. From this it can be concluded that the model is not as good at predicting new data and could indeed be improved, possibly by using another model.

Prediction of a new data observation

For each new classification, the distance to the K nearest neigbours are calculated (here in terms of the Euclidian distance), for K samples closest to the new datapoint, the class with the most occurences is chosen. The choice of a small values of K makes the classifier sensitive to noisy data, on the other hand a large K gives the risk of always choosing the most frequent class in the dataset. Another way to adress the problem with a small values of K would have been by weighing the points by the inverse of its distance $weight = \frac{1}{dist}$.

2.3 Decision Tree

As stated previously, by classifying with decision tress, we can split the data into different groups and map observations about an item (a man) to his target value (coronary heart disease or not). Since our target variable can only take a finite set of values $\{0, 1\}$ our model is a classification tree, not a regression tree, as this would have needed continouos variables.

By using the methods learned in this course, we will be able to choose the most balanced splits, which gives the lowest impurity og the branches and in turn will also help us identify which variables that have the highest impact on determining whether a person has coronary heart disease or not.

By using Hunt's algorithm, the full decision tree was created as shown below. Here, the maximum depth was 15 with `entropy` as split criterion and 14 using `gini`. This has been included below only to give an understanding of the size of the tree if one do not make any attempt to *prune* the tree.

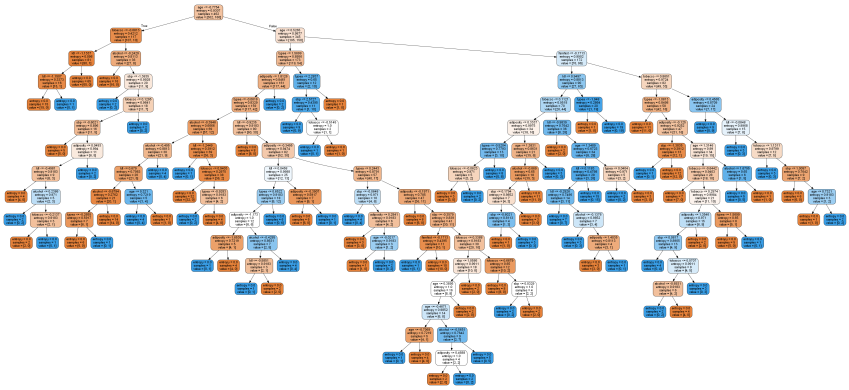


Figure 6: The full tree using Hunt's algorithm and ´entropy´ as the split criterion.

This algorithm for creating splits only stops when a pure split is encountered (when the impurity of the individual split is 0). Thus, this tree is of course highly overfitted and will not perform well on new data. It would be preferred to stop the growing of the tree and it's complexity by introducing a stop criterion. In order to find the optimal split, *pruning* was introduced with *cross-validation* to be able to take an informed decision on when to stop splitting. When deciding upon which purity measure to use both `gini` and `entropy` was tested.

Pruning to create the optimal decision tree was done using K-fold two-layer cross-validation, $K = 10$. For each new run of the K-fold cross-validation, the splits of data is shuffled. After running a few different times with different numbers of folds, $K \in \{5, 20, 50, 100\}$ it was made clear that increasing the number of splits did not make a significant difference in the results.

Increasing the maximum tree depth neither did improve the results – but since it was made clear that a depth of at most 16 would suffice in order to correctly classify all observations, this of course makes sense – even though the splits and data material might not be the same as for the entire tree do to pruning.

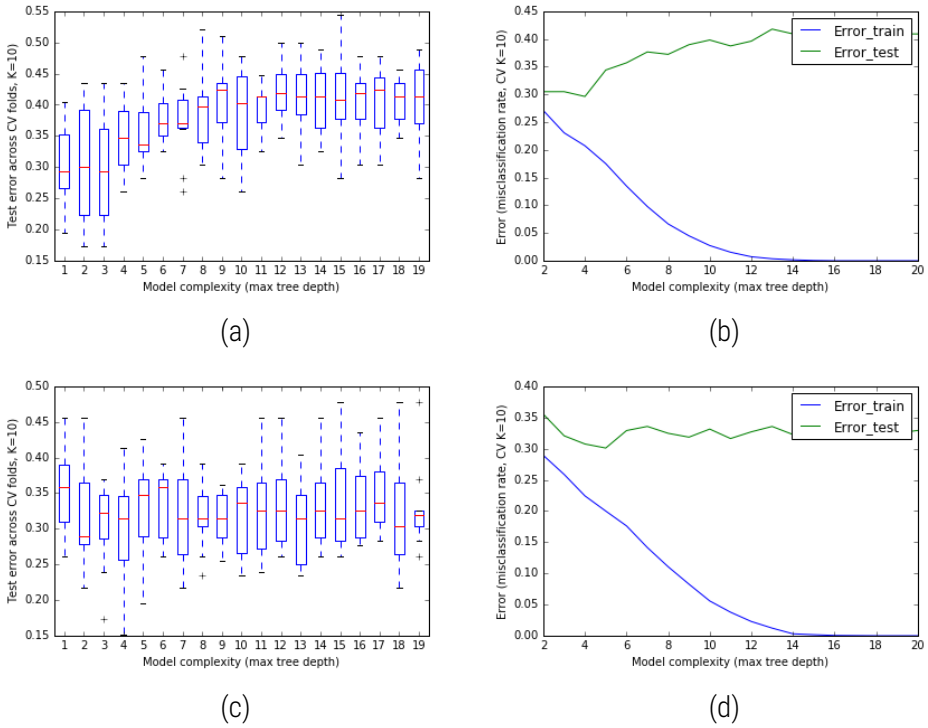


Figure 7: Figures a and c: Box plot of test error for each tree depth from 1-20 with split criterion 'gini' (top) and 'entropy (bottom). Figures b and d to the right: Model complexity and misclassification rate using 10-fold cross validation plotted for training and test set with split criterion 'gini' (top) and 'entropy' (bottom).

Loooking at the training and test error for the two plots, one cannot say much about the difference between the two impurity measures from the right-most plots alone. Here both plots reveal that the training error rapidly decreases with the tree depth, the test data however does not follow this trend. The lower right plot do have values in a slightly lower range $(0.30 - 0.35)$ than the upper left $(0.30 - 0.42)$. When looking at the boxplots it is made clear thath the mean test error generally is smaller using entropy as split criterion. These facts combined indicates that the best performance is acheived using a maximum tree depth of five. Thus this is the parameter used when pruning the decision tree.

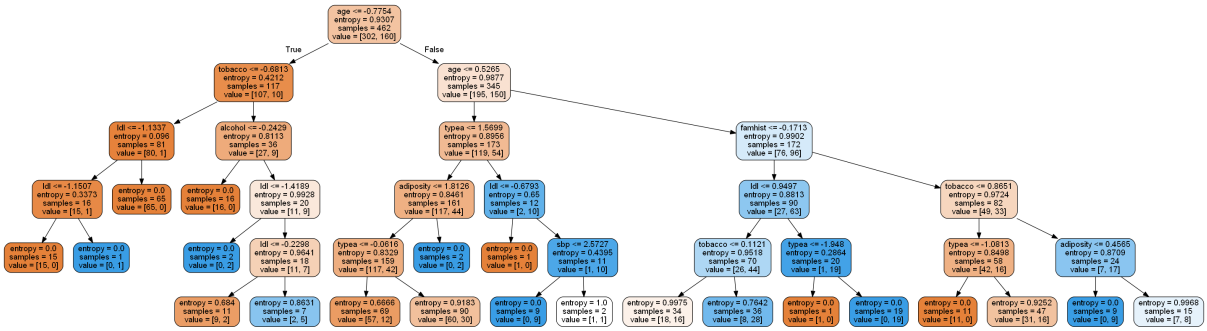


Figure 8: The Graphviz representation of the optimal decision tree with maximum depth of 5 and ´entropy´ as the split criterion.

Taking a closer look at the decision tree, the root of the tree splits on age. Thus, age is one of the important factors along with tobacco which is the next split to the left when prediction whether a person has coronary hearaaaaa. Different from the full tree, the entropy at each leaf is no longer 0. This is of course because of pruning and is the reason for the performance of the model on the test data.

Prediction of a new data observation

When predicting a new observation, the observation starts at the root of the corresponding decision tree following the split that most correctly classifies the given feature until a leaf has been reached. This means that each of the most important features of each observations is matched to find the best fit based on the given model.

2.4 Logistic regression

In logistic regression probability of the response taking a particular value is modeled based on combination of values that can be taken by the predictors. When using logistic regression it is important that there are no outliers and no high intercorrelation between tha data. This was taken care of in Project 1.

For the logistic regression, 10-fold cross-validation was used, with a changing parameter for the regularization strength, C. The test error did not vary across the different values for the regularization strength and are therefore not included in the report. The error rate was 26.62% which is relatively high – the highest number seen for the different classifiers, however, it is still better than randomly guessing. The error rate is accepted since we are training and testing on two different datasets, which means that the error rate is not overly optimistic as it would have been had we trained and tested on the same data. The model could have been optimized by using ROC-curves to make the optimal classification.

Prediction of a new data observation

Predictions in logistic regression can be compared to those of least-square regression. When predicting an observation, the prediction given by the logistic regression is of the probability p that the response variable y is 1, such that $p = P(Y = 1)$, this is not the same as predicting the value of Y, since Y in this case only can take one of the two values 0 and 1. Therefore a prediction could be $Y = 0.6$ which would mean that there is a probability of 60% that Y is in fact 1. Logistic regression fits a linear model to the log odds. This means that the equation show how to calculate the probability p form the input variables X and the estimates of the the regression coefficients.

2.5 Statistical comparison of performance

Comparing the models show that the smallest error rate is found for KNN and Logistic Regression. These models will be compared with the paired t-test.

Model	Error rate
KNN	24%
DT	29%
LR	26.62%

Table 3: Error rate for the KNN, Decision Tree (DT) and Logistic Regression (LR) classifiers.

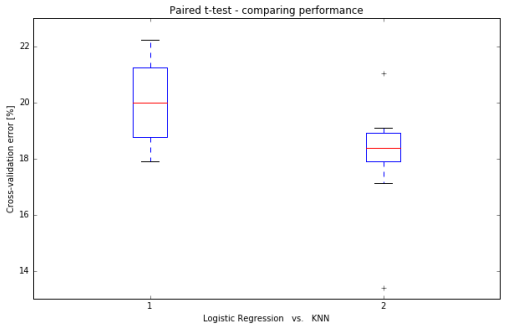


Figure 9: Comparison of the cross-validation error of models logistic regression and KNN

As can be seen in figure 9, KNN actually performs best on the data. The paired t-test indicates that the models are significantly different, with KNN as the best performer. Overall the models to not perform great on the test data.