**Application of an Artificial Intelligence System to Detect and Treat Heart Disease**

|  |  |
| --- | --- |
| Ayman Mezghani  ayman.mezghani@epfl.ch | Victoria Adcock  victoria.adcock@epfl.ch |

**TASK 1**

We used training data which had previously been discretized into bins, in order to obtain a model to predict whether a patient has heart disease. By implementing the Iterative Dichotomiser 3 (ID3) algorithm, we were able to successfully generate a decision tree with 70 leaves (see figure.1). ID3 is a *top-down greedy* algorithm which repeatedly divides features into two or more groups at each step. It starts building from the root node, and the feature with the greatest information gain is then selected at each iteration, as it calculates how well a given feature separates or classifies the target classes. Our model starts the classification process by separating the patients based on their chest pain type (cp). Depending on the patients' answers, their information will be passed through the appropriate subtree, and thus, the process continues until a classification is determined. At most, the tree will consider seven attributes in order to classify a patient as having/ not having heart disease (the height of the tree is 7), and a minimum of two attributes will be considered. Our model will consider on average 4.8 attributes before a classification can be determined.

A close up of a map

Description automatically generated

Figure 1: Decision tree generated after performing the ID3 algorithm on training data, which had been discretized into bins.

**TASK 2**

A screenshot of a cell phone

Description automatically generatedOnce our decision tree had been generated, it was necessary to test the accuracy of the model on a similar test dataset. Our model was able to correctly classify a patient for heart disease with an accuracy of 56.25%; 45 out of the 80 patients were correctly classified. As determined by the resulting confusion matrix (see figure. 2), our model has high precision and specificity (with values of 0.94 and 0.86 respectively). The high precision indicates that the majority of those who were classified as having heart disease, did in fact have heart disease. Likewise, the high specificity shows that the majority of people who did not have heart disease were correctly classified as such. However, our model has low sensitivity (0.50). This could be fatal in a real-world context, as this indicates that only half of the patients with heart disease were correctly identified as having heart disease, thus, the patients with heart disease who were misclassified as healthy would miss out on potentially lifesaving treatments. It would be preferable to optimise sensitivity in order to reduce the number of false negative cases, even if this would result in an increase in false positive cases.

Figure : Confusion Matrix for our decision tree model.

**TASK 3**

By analysing the training data, we were able to establish a list of *‘initial facts’,* as we already know which patients have heart disease, and their information regarding all the attributes recorded. Therefore, a list was created which contains all the information of the patients in the training set, where the *ith* element in the list is a list containing all the attributes, with their corresponding values, for the *ith* patient in the set.

By implementing the Depth-First Search (DFS) algorithm, we were able to find all of the paths in our decision tree, starting from the root node and terminating at the end nodes. We would use these paths to determine a set of *rules* stated by the tree; these rules explain the outcome of a classification. Since there are 70 end nodes, this resulted in 70 rules being established.

**TASK 4**

If we have a model which is able to predict whether a person has heart disease, it would equally be beneficial to be able to point to possible treatments. By separating the patients in the test data that were diagnosed as having heart disease, from the patients that were classified as healthy, and by tracing the path they followed in the decision tree, we are able find a potential treatment. As we had already established a set of rules for both positive and negative classification, we are able to use abduction to find a cause of heart disease and offer a solution. Since a person’s age and sex cannot be easily changed, nor can be a cause for heart disease, thus, cannot be considered faulty, these attributes were ignored when offering treatments. By comparing the rules for negative classification to the paths taken in the tree by the patients classified as having heart disease, we were able to find an attribute that leads to positive classification, and consequently highlight a change which results in negative classification.

The algorithm to find treatments successfully found a solution for all 35 patients who were diagnosed as having heart disease, according to our decision tree, with two or less attribute changes.

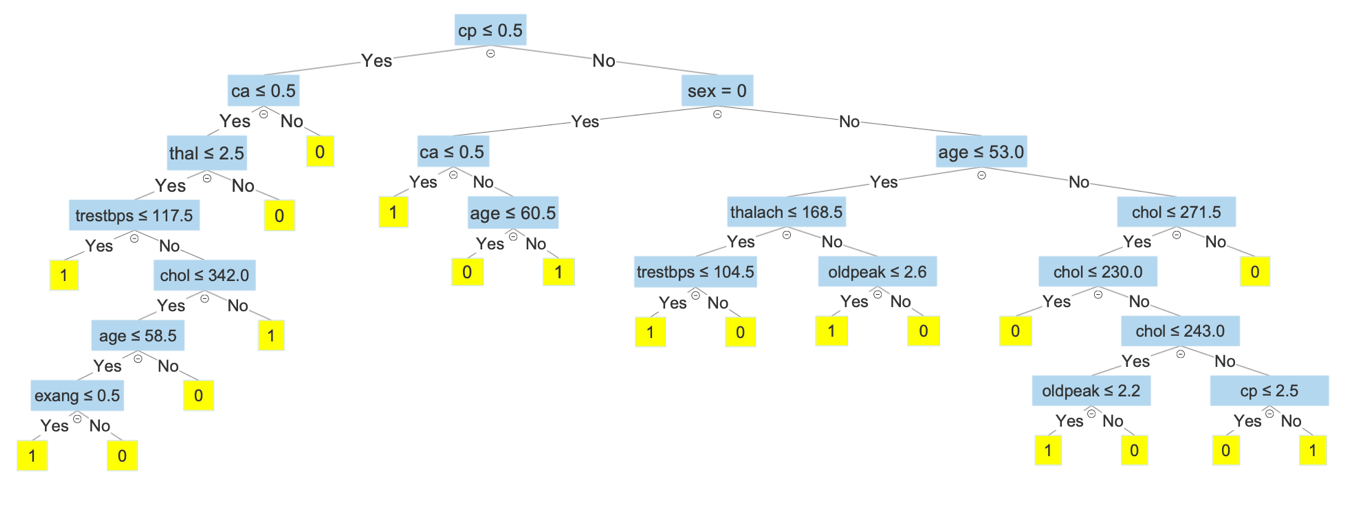
**TASK 5**

Figure : Updated decision tree using a continuous dataset.