Report for IKT 715 - Advanced Topics in Pattern Recognition*

Ayan Chatterjee[1]

University of Agder, Grimstad, Norway, ayan.chatterjee@uia.no

Abstract. This report, which has been written to fulfill the requirements of the course, **IKT 715 - Advanced Topics in Pattern Recognition**, identifies implement a few classification algorithms including the **Optimal Bayesian classifier**, **Anti-Bayesian classifier**, **Decision Trees (DTs)**, and **Dependence Trees (DepTs)**. First, a brief introduction is given about the report. In the second section, types of datasets used here are discussed. In the third section, comparison in between classification accuracies of the bayesian and anti-bayesian classification methods for the artificial and real-life data sets is highlighted. In the fourth section, comparison in between classification accuracies of the DT and DepT classification methods for the artificial and real-life data sets is discussed. In the fifth section, comparison in between classification accuracies of the four algorithms such as (a) Bayes, (b) Anti-Bayes, (c) DTs, and (d) DepTs for the artificial and real data sets is elaborated.

Keywords: Bayesian · Anti-Bayesian · Decision Tree · Dependence Tree.

1 Introduction

Pattern recognition is a process of identifying patterns by using machine learning algorithms. Pattern recognition can be specified as the classification of data based on knowledge already obtained or on statistical information separated from patterns and/or their interpretation. In a regular pattern recognition application, the raw data is processed and transformed into a form that is flexible for a machine to use. Pattern recognition encompasses classification and cluster of patterns. The objective behind pattern recognition algorithms is to deliver a sensible answer for all possible data and to classify input data into objects or classes based on certain features.

In machine learning, the entire dataset is divided into two categories, one which is used to train the model i.e. training set and the other that is used in testing the model following training, i.e. testing set. In this course, we have learnt a. how to develop algorithms for bayesian, anti-bayesian, decision tree and dependence tree, b. artificial data generation, c. train and test the same developed models against real and artificial datasets and followed by accuracy prediction and comparison.

* Organized by 'University of Agder'.

2 Data-Sets and Data Pre-Processing

In the entire study, we have used two types of data-set - a. real glass identification data-set b. artificial generated data-set.

2.1 Real Data-Set

The Glass identification data-set has been used to classify the type of glass, given the following features, specified in this order: 1. ID 2. RI: Refractive index 3. Na: Sodium 4. Mg: Magnesium 5. Al: Aluminum 6. Si: Silicon 7. K: Potassium 8. Ca: Calcium 9. Ba: Barium 10. Fe: Iron 11. Class (6 possible types). This data-set can be found at the UCI Machine Learning Repository (https://archive.ics.uci.edu/ml/machine-learning-databases/glass/).

Data pre-processing steps on real glass data:

Step-1: Load the data-set with python library (pandas/numpy)
Step-2: Remove **Id: Number** feature
Step-3: Check if any missing value is found [if true then perform cleaning]
Step-4: Shuffle the cleaned data to return a random sample of items from an axis of object.
Step-5: Partition the data for training and testing (80:20 random selection)
Step-6: Train a machine learning model (algorithm) with the training data-set and perform testing on the trained model with testing data-set to determine the classification accuracy of the model.

The shape of the real glass-data is (214, 11) which is distributed among 6 classes as follows: class - 1 (70, 11), class - 2 (76, 11), class - 3 (17, 11), class - 5 (13, 11), class - 6 (9, 11), class - 7 (29, 11). In this assignment, we have used class-1, class-2, class-3 and class-7 for multi-class classification (as that combination gives us the highest data) using bayesian, anti-bayesian, decision tree and dependence tree. For training and classifying using the DT and DepT, we have used binary features for the real-life glass data adopting a threshold mechanism. We have used **mean** value determination (as a threshold) from each features to generate binary data from real glass data. Values greater than the threshold, are converted to **1** otherwise converted **0**.

2.2 Artificial Data-Set with DepT

The artificial data-set is generated randomly based on the probability theory. Generated artificial data has 10-dimensional feature space (including index which is removed in the data pre-processing) with 4 classes. We have assumed that the vector components obey a **dependence tree** structure between the various features. This **DepT** has been arbitrarily assigned and unknown to the classification algorithm for training and testing. For each of the 4-classes and for each of the

10-features, we have randomly generated the probabilities of the feature taking the value **1** or **0**.

Rule: For **class j = 1 to 4** and for feature indices **i = 1 to 10**, we have randomly assigned the value **Vi,j = Pr[Xi = 0 W = Wj]** following a **DepT** structure where **P** is the probability. Based on the distribution, we have modelled 4-classes. Generation of each feature point is dependant to other feature point. They are unknown to the **pattern recognition (PR)** system, but known to the data generation program.

Example: For class W1, p1 = [X1 = 0 | X3 = 0; W1] = 0.7, $p1 = [X1 = 0 \mid$ X3 = 1; W1] = 0.15, and for the class W2, q1 = [X1 = 0 | X3 = 0; W2] = 0.9, $q1 = [X1 = 0 \mid$ X3 = 1; W2] = 0.27, where p, q are the probabilities related to classes W1 and W2 and X are the features. Each generated feature point is dependent to another.

Assumed dependence tree: Our assumed decision tree has a feature vector [X0, X1, X2, X3, X4, X5, X6, X7, X8, X9]. We have generated the root node (X3) first and followed by other nodes following the above rule. Node X1, X5, X7 depends on X3. Node X6, X9 depends on X1. Node X2 depends on X5. Node X0 depends on X8, and node X4, X8 depends on X7. We have generated total 8000 artificial data samples for 4 classes (2000 data-sample / class) following the assignment requirement based on the feature dependency. We have performed **classification** assuming that all the random variables are dependent based on the DepT.

2.3 Artificial Data-Set with Feature In-dependency

The artificial data-set is generated randomly based on the probability theory following the rule: P[Xi = 0 | W = W1] = pi, [Xi = 1 | W = W1] = 1 - pi, and [Xi = 0 | W = W2] = qi, [Xi = 0 | W = W2] = 1 - qi, where p, q are the probabilities related to classes W1 and W2 and X are the features. Each generated feature point is independent to another. They are known to the **pattern recognition (PR)** system, but known to the data generation program.

We have generated total 8000 artificial data samples for 4 classes (2000 data-sample / class) following the assignment requirement based on the feature in-dependency. We have performed a **classification** assuming that all the random variables are independent following a binary distribution.

2.4 Bayes Classification Rule for Binary Data

Binary data does not follow gaussian/normal distribution. It has different classification rule:

G(X) = $\sum_{i=1}^{d} X_i$ [log ((1-p_i)/(1-q_i) - log(p_i/q_i)] + log (P1/P2) + $\sum_{i=1}^{d}$ log(p_i/q_i), if G(X) > 0, then

$$X \in W1 - class$$

else

$$X \in W2 - class$$

Where, P(X | W1) = $\Pi_{i=1}^{d}\ p_i{}^{(1-X_i)}$.(1-$p_i$)$^{(X_i)}$, P(X | W2) = $\Pi_{i=1}^{d}\ q_i{}^{(1-X_i)}$.(1-$q_i$)$^{(X_i)}$, X is data sample and d is total number of features.

2.5 Training and Testing Strategy

With regard to training and testing, we have used 5-fold cross-validation (for granulation by removing little randomness) to evaluate machine learning classification models on the limited data samples both real and artificial.

2.6 Multi-class Classification Technique Adopted

We have adopted **binary class comparison** method to calculate the final winning class. The resultant **confusion matrix (CM)** has helped us to evaluate classification model accuracy.

3 Bayesian vs Anti-Bayesian

Bayesian classification is based on **Bayes' Theorem** that includes two types of probabilities - posterior probability [P(H/X)] and prior probability [P(H)], provided X is data tuple and H is some hypothesis or a class in a classification problem. The ultimate target is to create a classifier or a simple discrimination function ([P(H/X)] = P(X/H)P(H) / P(X)) based on the training data to classify provided testing data among trained classes. P(X/H) is called class conditional distribution. A good discrimination function gives good accuracy. A general bayesian classifier can be represented as below:

G(X) = (P(X/W1)P(W1) / P(X)) - (P(X/W2)P(W2) / P(X)), if G(X) > 0, then data tuple X will belong to class W1 else class W2.

Bayesian is optimal but costly and hard. We have implemented both optimal bayesian and naive bayesian following a gaussian distribution of provided real glass data. The derived classifier is quadratic in nature and is equated as below:

G(X) = $X^T \cdot A \cdot X + B \cdot X + C$, if G(X) > 0, then data tuple X will belong to class W1 else class W2. A, B, C are constants]

$$A = (\sum_B{}^{-1} - \sum_A{}^{-1})$$
$$B = -\ (2 \times (M_B{}^T \cdot \sum_B{}^{-1} - M_A{}^T \cdot \sum_A{}^{-1}))$$
$$C = (M_B{}^T \cdot \sum_B{}^{-1} \cdot M_B - M_A{}^T \cdot \sum_A{}^{-1} \cdot M_A + ln \mid \sum_B \mid - ln \mid \sum_A \mid)$$

Our data X has a feature dimension d = 9 and the distribution follows normal or gaussian following the central limit theorem. μ is the mean and σ^2 is the variance of the distribution. $\sum$ is the co-variance matrix (d × d). The only difference between **Naive Bayes** and **Optimal Bayes** classifiers is the co-variance matrix used. For Naive Bayes, $\sigma_i{}^2$ = E $[x_i - \mu_i]^2$ [diagonal matrix] and for optimal bayes, $\sigma_i\sigma_j$ = E$[(x_i - \mu_i)(x_j - \mu_j)]$ [full matrix].

The steps are explained as below:

Step-1: Load the real glass data-set and perform pre-processing as explained in section-2.1

Step-2: Partition the random data into 5-folds (4-folds for training and 1-fold for testing). For each fold, perform below step-3 to step-6:

Step-3: Partition the training data into four sets those belonging to four classes

Step-4: Calculate the Normal Distribution for each set (μ and σ)

$\mu = 1/\mathrm{N} \sum_{i=1}^{N_i} \mathrm{X}_i$ and $\sum^{+} = 1/(N-1) \sum_{i=1}^{N_i} (\mathrm{X}_i - \mu_i)\ (\mathrm{X}_i - \mu_i)^T$, where X is given feature vector and N is total number of data sample.

Step-5: Form the classification equation as demonstrated above (section-3)

Step-6: Perform binary class comparison for classification of test sample.

Step-7: After 5-fold cross validation, calculate average testing accuracy.

Note: We have used pseudo-determinant and pseudo-inverse for covarience calculation to avoid determinant of a singular matrix (0), and the logarithm of 0 is undefined.

Anti-Bayesian statistical pattern recognition is another classification technique. We have used the same approach in real glass data-set assuming gaussian distribution, where classification is performed by quantile statistics (pattern recognition based on the two points rather than mean).

The steps for Anti-Bayesian are explained as below:

Step-1: Load the real glass data-set and perform pre-processing as explained in section-2.1

Step-2: Partition the random data into 5-folds (4-folds for training and 1-fold for testing). For each fold, perform below step-3 to step-6:

Step-3: Partition the training data into four sets those belonging to four classes

Step-4: Find two points (1/3rd percentile: a_i) and (2/3rd percentile: b_i) for each class considering gaussian assumption.

Step-5: Perform classification of test sample (x) based on the rule: if $\mathrm{x}^* < b_1$ and $\mathrm{x}^* > \mathrm{a}_1, x^* \in$ W1 and if $\mathrm{x}^* < b_2$ and $\mathrm{x}^* > a_2$, $\mathrm{x}^* \in$ W2. If, $b_1 < \mathrm{x}^* < a_2$, then decision is based on the: $||\ \mathrm{x}^* - \mathrm{b}_1||$ and $||\ \mathrm{x}^* - \mathrm{a}_2||$. [This process is performed for each features of the test sample for future majority vote calculation]

Step-6: Perform binary class comparison for classification of test sample following majority voting scheme.

Step-7: After 5-fold cross validation, calculate average testing accuracy.

3.1 Real Glass Data

Optimal Bayes	Class-1	Class-2	Class-3	Class-7	Accuracy
Class-1	37	0	6	10	61% (rounded)
Class-2	13	26	7	17	
Class-3	1	0	13	0	
Class-7	0	0	6	16	

Naïve Bayes	Class-1	Class-2	Class-3	Class-7	Accuracy
Class-1	17	40	3	9	60% (rounded)
Class-2	2	62	0	12	
Class-3	1	8	8	1	
Class-7	0	1	0	27	

Anti-Bayes	Class-1	Class-2	Class-3	Class-7	Accuracy
Class-1	49	11	2	6	42% (rounded)
Class-2	52	13	2	9	
Class-3	12	2	0	3	
Class-7	6	5	1	17	

Fig. 1. Confusion matrix (for 5-folds) and average accuracy comparison in between optimal bayes, naive bayes and anti-bayes for real glass dataset (4-class).

The given real glass dataset is small. Optimal Bayes gives the highest accuracy than naive-bayes and anti-bayesian. In anti-bayesian with gaussian assumption,and followed by quantile statistics seems some information loss.

3.2 Artificial Data-Set

The artificial dataset (8000×11) is generated and classified as described in section- 2.2 and section- 2.3. As depicted in **fig-2**, **Bayes-I** represents bayesian classification based on the independent random variables and **Bayes-II** represents bayesian classification based on the dependent random variables (based on the DepT that has been inferred). For random data, we have assumed a binary distribution instead of gaussian distribution for the features.

It is not possible to perform anti-bayesian classification over artificial binary data. The binary data is non-gaussian and its histogram will produce only two columns (0 and 1). Over binary data, quantile points will be hardly able to classify features or attributes.

Bayes - I	Class-1	Class-2	Class-3	Class-4	Accuracy
Class-1	2000	0	0	0	25% (rounded)
Class-2	2000	0	0	0	
Class-3	2000	0	0	0	
Class-4	2000	0	0	0	

Bayes - II	Class-1	Class-2	Class-3	Class-4	Accuracy
Class-1	1998	0	0	0	25% (rounded)
Class-2	1999	0	0	0	
Class-3	2000	0	0	0	
Class-4	1998	0	0	0	

Fig. 2. Confusion matrix (for 5-folds) and average accuracy of bayes for artificial dataset (4-class).

4 Decision Tree vs DepT

A decision tree makes a decision based on a tree representation with a set of if-else rule. Each node in the tree specifies a test. It is used for approximating discrete valued target function (non-gaussian). Tree is created with recursion based on apriori data. **Fig-7** represents a sample decision tree created from real-glass data after rendering to binary and following the algorithm as described below.

The steps for Decision Tree are explained as below:

Step-1: Load the real glass data-set and perform pre-processing as explained in section-2.1 [here, rendered to binary data based on the threshold]
Step-2: Calculate the entropy for entire data-set
Step-3: for every feature do perform the following steps:
1.calculate entropy for all categorical values
2.take average information entropy for the current attribute
3.calculate gain for the current attribute
Step-4: Select the highest gain attribute.
Step-5: Repeat until we get the desired tree.

Dependence tree model contains a dependence in between nodes (features) which is unknown to the pattern recognition (PR) system. Generated each feature point is dependent on another (parent node). Data generation with DepT is explained at section-2.2. Probability of every node depends on its parent.

The steps for DepT are explained as below:

Step-1: Load the real glass data-set or artificially generated data-set and perform pre-processing as explained in section-2.1 [here, rendered to binary data based on the threshold]

Step-2: Create a fully connected undirected graph (G). G = (V,E,W)
V = set of vertices (random variables)
E = set of edges (connecting any two vertices V_i and V_j)
W = set of weights associated to each edge

Step-3: Calculate weight (W) in graph (edges) following **Expected Mutual Information Measure**

Step-4: Root probability calculation for competing classes (say, W1, W2) for particular test sample and followed by, calculate P_a(V) = **probability for first and second order marginals**

Step-5: Class with better P_a(V), will be the fittest

4.1 Real Glass Data (Rendered Binary)

DT	Class-1	Class-2	Class-3	Class-7	Accuracy
Class-1	59	19	10	3	71% (rounded)
Class-2	7	50	5	3	
Class-3	0	4	2	0	
Class-7	2	3	0	23	

DepT	Class-1	Class-2	Class-3	Class-7	Accuracy
Class-1	32	15	15	7	39% (rounded)
Class-2	23	21	20	11	
Class-3	7	0	7	3	
Class-7	10	2	3	14	

Fig. 3. Confusion matrix (for 5-folds) and average accuracy comparison in between DT and DepT for real glass dataset (4-class).

The used glass data sample size is small. Obtaining a good accuracy with this dataset is a challenge. Binary conversion from real number has a chance to loose information. Training with more data can increase the prediction accuracy. The rendered binary data used to train DepT is not generated based on a dependence tree and thereby, accuracy is low.

4.2 Artificial Data

Increasing the volume of the training data with the artificial data generation technique based on the dependence tree (section-2.2) we have increased the classification accuracy both in DT and DepT as depicted below:

DT-Artificial	Class-1	Class-2	Class-3	Class-4	Accuracy
Class-1	1806	136	212	64	79% (rounded)
Class-2	77	1496	288	112	
Class-3	94	242	1339	161	
Class-4	22	124	159	1663	

DepT-Artificial	Class-1	Class-2	Class-3	Class-4	Accuracy
Class-1	1408	82	260	248	51% (rounded)
Class-2	562	239	488	710	
Class-3	317	126	1132	425	
Class-4	113	18	602	1265	

Fig. 4. Confusion matrix (for 5-folds) and average accuracy comparison in between DT and DepT for artificial dataset (4-class).

5 Accuracy Comparison

5.1 Real Glass Data:

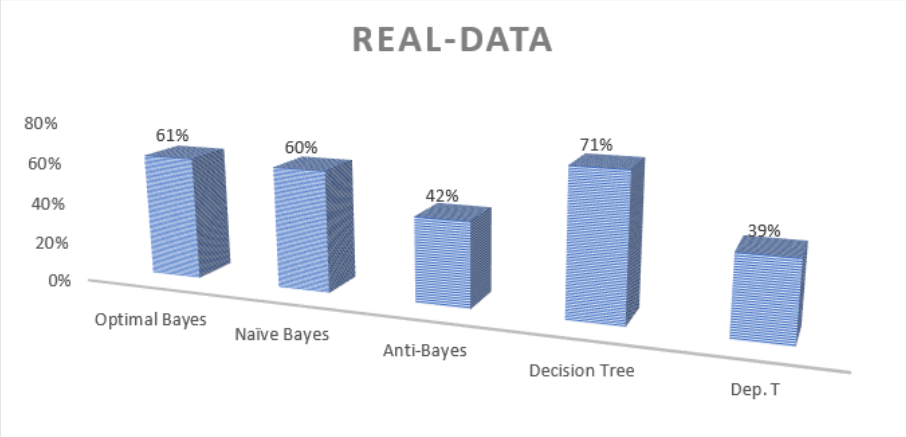

Fig. 5. Accuracy comparison for real data

5.2 Artificial Data

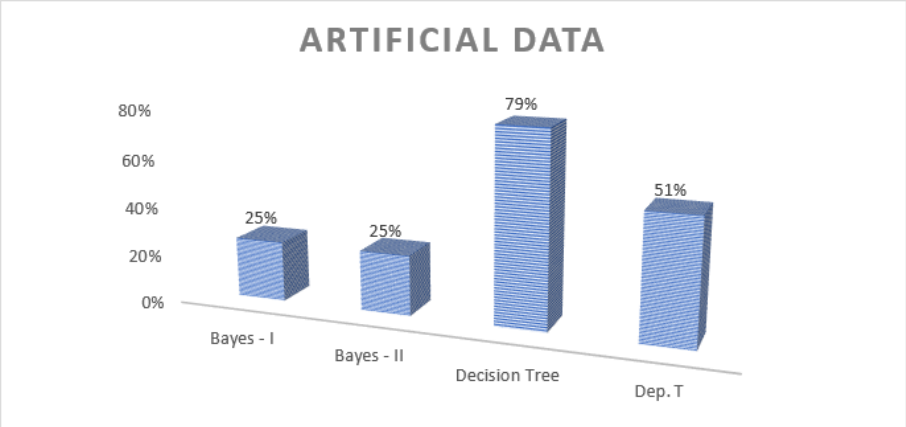

Fig. 6. Accuracy comparison for artificial data

5.3 Explanation

The provided real glass data is very small for training a pattern recognition model in order to achieve higher accuracy. Some of the features have mostly zero values, but we have considered those features as well in our training else data volume would have become even small. As a consequence, it hits the accuracy calculation. Instead of real-inverse and determinant, we have calculated approximated value (pseudo-determinant and pseudo-inverse) in covariance calculation for Bayesian.

As depicted in **fig-5,** it seems that with limited set of real data, Bayes has performed well. Anti-bayesian produces less accuracy on limited data as compared to Bayes. We have made a gaussian assumption for anti-bayesian and there might be some information loss due to quantile calculation and missing-out border point data. Histogram might produce better approximation but we have not tried it here. Decision tree has produced the most significant accuracy with this data after rendered them to binary. Binary conversion from real number has a chance to loose information. Training with more data can increase the prediction accuracy. The binary data fed to DepT is not generated based on a dependence tree and thereby, accuracy is low for DepT.

As depicted in **fig-6, Bayes-I** represents bayesian classification based on the independent random variables and **Bayes-II** represents bayesian classification based on the dependent random variables (based on the DepT that has been inferred).

Generated artificial data is binary and they are not following gaussian distribution. Artificial data for bayes has been generated randomly as specified in the section- 2.2 and section- 2.3. The performance of bayes with binary data is not good due to randomness in data. On the other hand, it is not possible to perform anti-bayesian classification over artificial binary data. The binary data

is non-gaussian and its histogram will produce only two bins (0 and 1). Over binary data, quantile points will be hardly able to classify features.

Artificial data generated by feature dependence (DepT) has helped to increase the accuracy of DT and DepT as compared to the real glass data.

6 Conclusion

It seems that if we can reduce some assumed randomness in artificial data generation, we can increase the accuracy of the classifier function. More data in training will increase the accuracy of the classifier. For real life data, we must select more robust dataset for proper training.

Thanks to Dr. John Oommen (oommen@scs.carleton.ca) for teaching us some advanced concepts on pattern recognition as summarized in this report. Thanks to Rebekka Olsson Omslandseter (rebekka.o.omslandseter@uia.no) for supporting us during the course and reviewing the source code written in python.

In the next page, a sample decision tree is depicted (drawn with grpahviz) (**fig-7**) with binary decision making on every branches. Calculated entropy is displayed at each winning node. Leaf nodes are the classes.

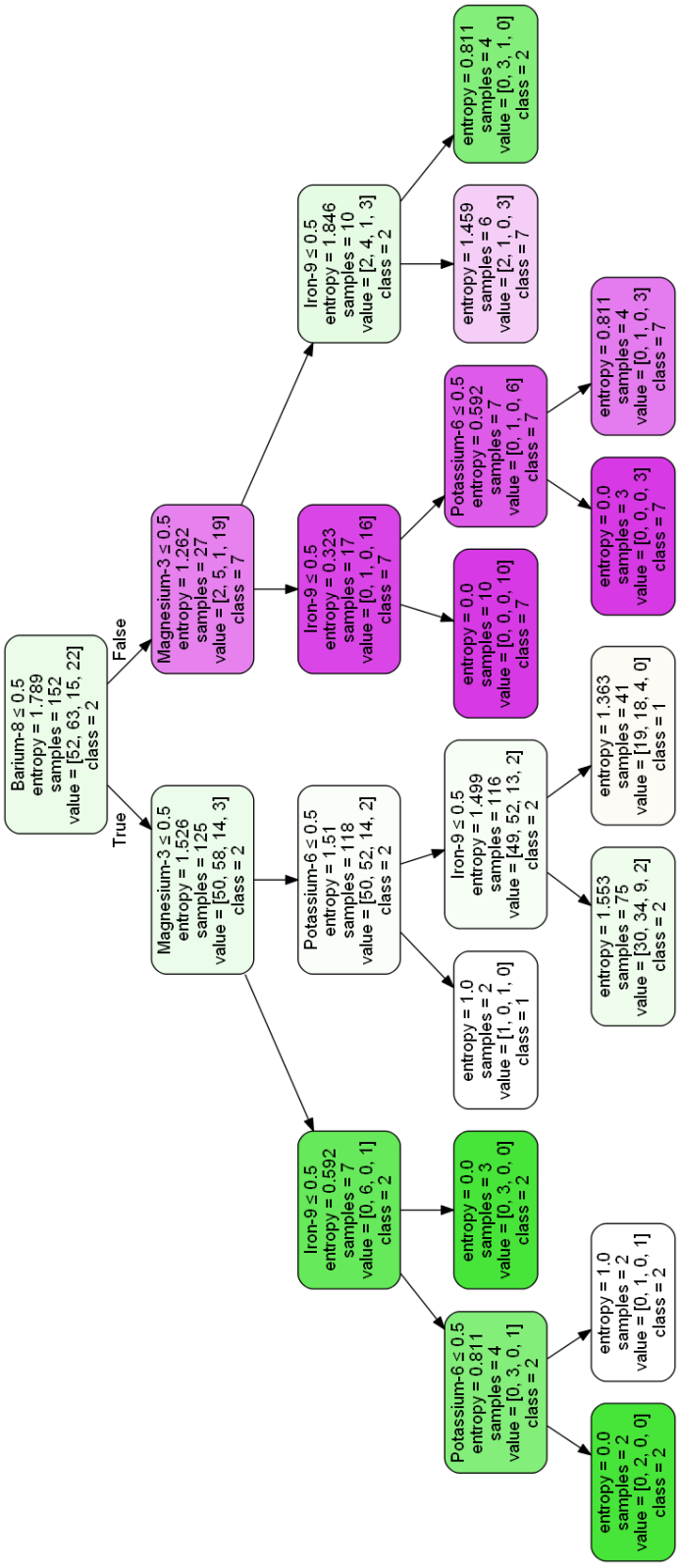

Fig. 7. A sample Decision Tree