6G6Z1705

Artificial Intelligence

Scenario 2

14032908
Joshua Michael Ephraim Bridge
joshua.m.bridge@stu.mmu.ac.uk

April 16, 2018

1 Introduction

In this report an AI classifier will be put forward which maps mamographical data to desired outputs (diagnoses). In order to do this two types of AI classifiers will be evaluated on their performance in this task, along with relevant pre-processing of the attributes to enhance classifier performance. The two classifier types will be a Decision Tree (J.48) and an Artificial Neural Network (Multilayer Perceptron, Minsky et al. (2017)). In order to evaluate their performance, considerations of both training time & classification accuracy will be taken into account.

2 AI classifiers

In this section a brief study will be conducted into the two classifier types mentioned previously.

2.1 Decision Trees

A decision tree is a type of classifier (specifically a hierarchical variant of a multistage classifier, as defined by Safavian & Landgrebe (1991)) which uses a tree-like structure to test values on different attributes in a format similar to a flow chart. The tree structure itself could be described as a single root node with 0 to many connected children, each themselves with 0 to many connected children. Any node in a decision tree with no children is known as a leaf node and has a direct relationship with a class label. At each node in the tree a test is carried out on an attribute and the result of that test decides on which of the child nodes the process should continue onto. The process of completing each test from the root node to a leaf node should result in a classification of the data provided.

Self-learning decision trees are often very useful because they explicitly define how the instances are classified within the tree, simplifying the process into a set of simple rules. This is different to ANN's (see section 2.2) where the classification process is mostly hidden and can often be a very complex set of rules which would be very hard to follow.

2.1.1 J.48 Parameters

Within this report the J.48 decision tree algorithm will be used (an implementation of the C45 decision tree by Quinlan (2014)). Below are the parameters which will affect classifier performance.

Confidence. The confidence parameter is a way of controlling the amount of error-based pruning (Quinlan 1987) within the decision tree. More specifically, post-pruning is the process of estimating the error rate (probability of mis-classification) at each node in the tree, and deciding whether or not to remove the node. Lower values of the confidence factor will result in the post-pruning becoming much more aggressive with removing nodes (Beck et al. 2008).

Minimum number of objects. Within weka the Minimum number of objects parameter controls the Minimum number of instances per leaf. This means that each leaf within the tree must have at least the specified amount of classified instances for it not to be pruned. This parameter is good for data-sets which are particularly noisy which could introduce some leaf nodes which are not very stable classifiers. With a higher minimum number of objects, the tree will likely become much more pruned.

2.2 Artificial Neural Networks

An Artificial Neural Network is a mathematical system which is able to classify data by performing a series of mathematical functions (activation functions) which take weightings for each of their inputs and summarise them into a single output. ANN's are designed in light of the way human/animal brains process information, via a series of neurons which are connected (in biology these connections are called synapses). Within an ANN each neuron is connected to either input attributes or the output of neuron(s) in another layer of the network. The connections between the neurons contain weightings which is the main principal behind how the network can emphasise some data over others.

The neurons within an ANN can be split up into a series of 'layers', where the outputs from one layer of neurons will become the inputs for the next layer of neurons. Within a Multilayer Perceptron (see section 2.2.1) the layers consist of 1 input layer, 1 output layer, and at least 1 'hidden layer' where each neuron in the hidden layer(s) and the output layer

are neurons which perform an activation function.

Within an ANN there must be a process of 'learning' which enables it to find the most optimal values for the weights which are used in the activation functions. This is done via propagation which enables the algorithm to modify the weights based on the error rate of the output, compared to the expected output.

2.2.1 Multilayer Perceptron parameters

Within this report the Multilayer Perceptron (Minsky et al. 2017) variant of ANN will be used and below are the parameters which will affect classification performance.

Hidden Layers. The hidden layers parameter allows the user to define the structure of the network they would like to train. Introducing more layers & neurons introduces more complexity which is good for more complex datasets with attributes which are not linearly separable, however for simpler datasets this may introduce unwanted complexity within the network. What hidden layers are and how they relate to the ANN is explained in more detail in section 2.2.

Learning Rate. Learning rate applies to the propagation algorithm and more specifically the Gradient Descent. It concerns the speed at which the minimum squared error is reached. A low learning rate would mean that many updates (a high training time) would be needed in order to find the global minimum - which is not desirable. If the learning rate is too high however, then this can lead to divergent behaviour where the propagation algorithm is not able to correctly settle on an optimal minima.

Momentum. Once again momentum relates to the gradient descent for squared error, however momentum defines the way in which the minima is reached. As there may be several local minimas within the descent path, it would not be desirable to end in a minima which is not actually the global minima. In order to avoid this, the momentum value is linked to the learning rate in that increasing the momentum allows the descent path to continue past local minimas in search of a lower squared error.

Training Time. The main factor in the learning process of an ANN is the amount of time it has to train. There is no point in time which the network will be 'done' learning

therefore the most optimal amount of learning time must be chosen. Training time is measured in 'epochs', where 1 epoch is the completion of a single training iteration. A training iteration includes the inputs passing through every layer, providing an output, and then the propagation algorithm updating the weights and biases for each applicable neuron. If the training time is too high, this can lead to something called 'over-training' where the ANN becomes too dependent on the training data and will start giving worse results when presented with unseen testing data. Therefore it is necessary to find the optimal training time, in combination with the optimal learning rate and momentum.

3 Data set analysis

In this section, the mamographical dataset will be analysed as a preparatory step to pre-processing of the data.

3.1 BI-RADS

3.1.1 Measurement

The acronym BI-RADS stands for "Breast Imaging Reporting and Data System" (American College of Radiology 1998). It is a system which was designed to introduce some standardisation into the field of diagnosing breast cancer. The score for BI-RADS is on an ordinal scale from 1 to 5, with 1 being benign and 5 being very likely malignant. Below are the 5 definitions of the BI-RADS scale as defined by American College of Radiology (1998).

1) Negative

2) Benign findings

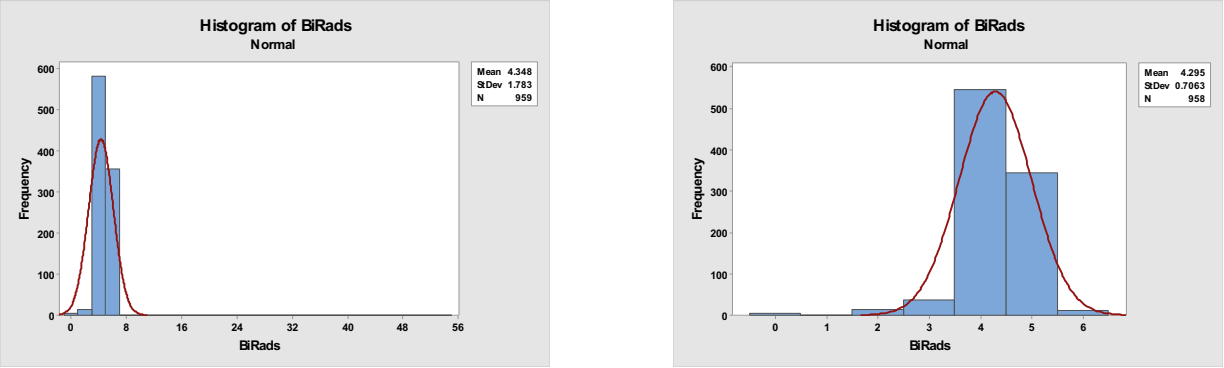
3) Probably benign

4) Suspicious abnormality

5) Highly suggesting of malignancy

3.1.2 Distribution

As shown in figure 1, the distribution of BI-RADS scores is non-normal and is skewed to the right.



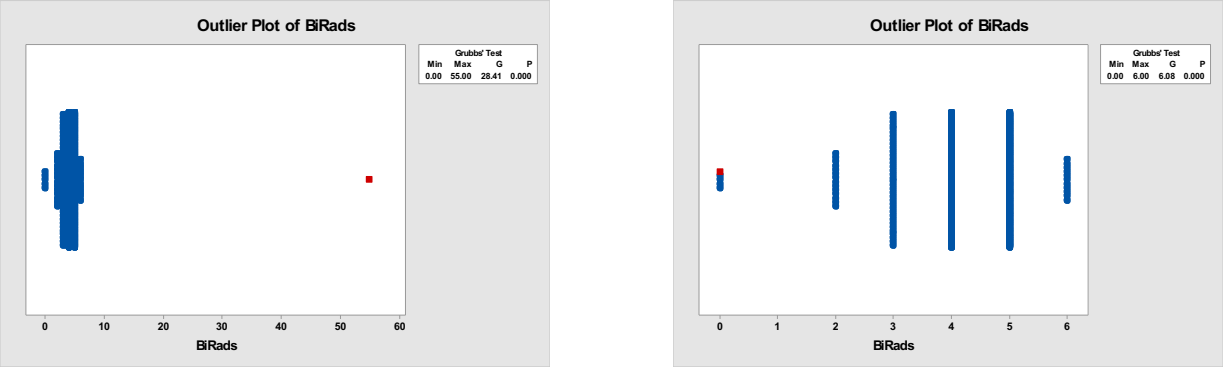
(a) Initial (b) Outlier replaced with median

Figure 1: BI-RADS histogram

3.1.3 Outliers

Within the BI-RADS attribute there is at least one major outlier as shown in figure 2a. This outlier value is '55' and while it could be guessed that this is due to human error of entering a score of '5' twice accidentally, this outlier must be replaced by the central tendency. This outlier plot has been repeated in figure 2b, with the outlier replaced with the median so as to get a clearer indication of other outliers.

Figure 2b shows that there are several instances with BI-RADS scores of both '0' and '6' which do not exist in the scale. These scores do exist in edition 4 of the BI-RADS scale (D'orsi et al. 2003) however the data descriptors provided with the dataset make no mention of these categories or which edition of the BI-RADS scale it refers to.



(a) Initial (b) Outlier replaced with median

Figure 2: BI-RADS outlier plots

3.1.4 Predictive

With the BI-RADS score being a predictive scale, this would mean that if the score of BI-RADS was high then the chance of that instance having a malignant severity would be much higher. This can be shown in figure 3 where it is shown that there is a slight correlation between a higher BI-RADS score and % of severity classification.

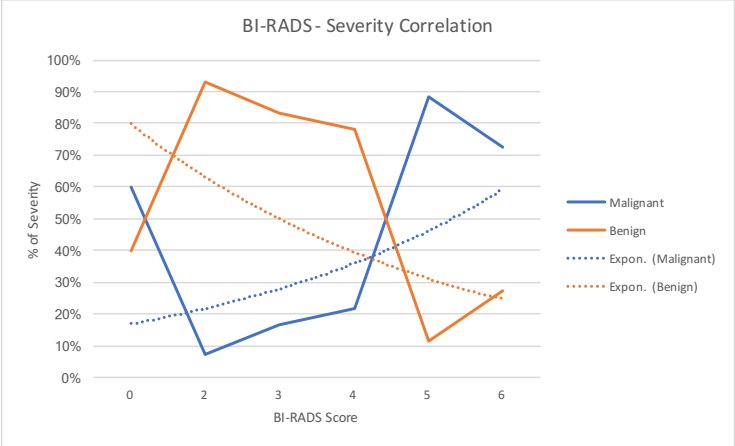
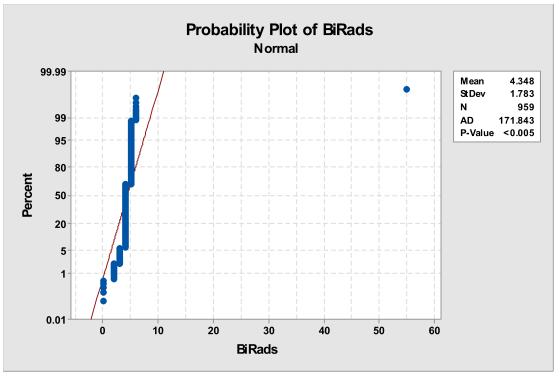
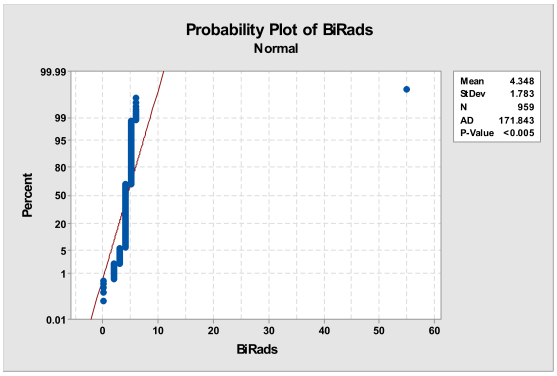


Figure 3: BI-RADS - Severity correlation



(a) Initial



(b) Outlier replaced with median

Figure 4: BI-RADS probability.

3.2 Age

3.2.1 Measurement

This is the only non-predictive attribute provided in the dataset, and is a simple ratio scale of the patients age.

3.2.2 Distribution

As shown in figure 5, the distribution of age within the dataset is non-normal and is skewed very slightly to the right.

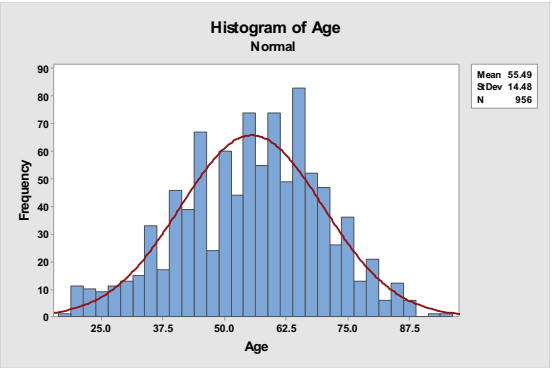


Figure 5: Age histogram

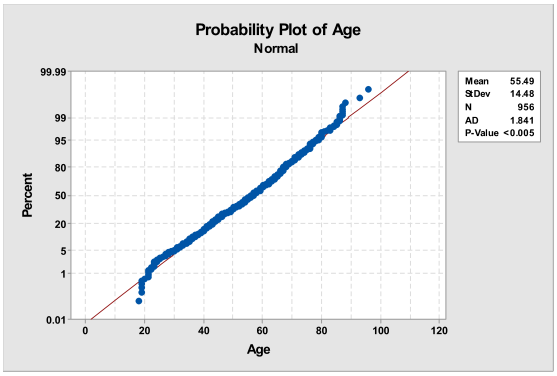


Figure 6: Age probability

3.2.3 Outliers

There does not appear to be any outliers in the age attribute as shown by figure 7. There are, however, 2 missing values within the attribute. The central tendency for this attribute should be the median due to the fact that the scale type is ratio and that the distribution is non-normal.

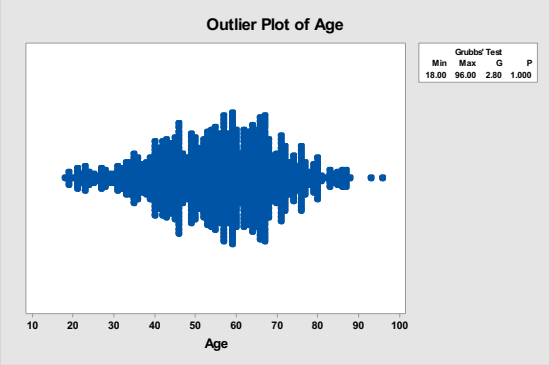


Figure 7: Age outlier plot

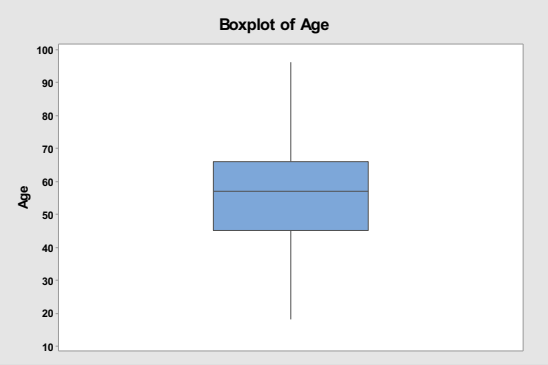


Figure 8: Age boxplot

3.2.4 Predictive

While age itself is not a predictive attribute, it has been shown that women diagnosed with breast cancer are much more likely to be above 50 years of age (Kerlikowske et al. 1993). This can be backed up by figure 9 which shows a clear correlation between age and the percent of severity classifications per age group in the provided dataset

Due to this and the findings with the BI-RADS attribute, it could be inferred that those with an age above 50 and a high BI-RADS score have a very high change of having a malignant severity.

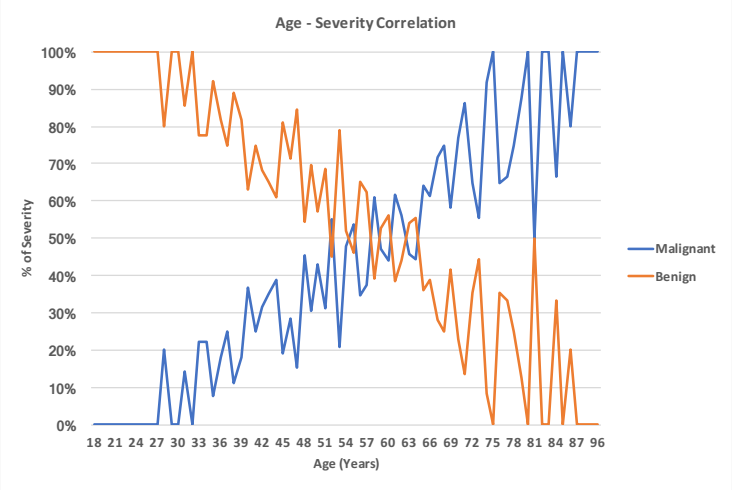


Figure 9: Age - Severity correlation

3.3 Shape

3.3.1 Measurement

This attribute is used to describe the shape of the mass being investigated. It is a nominal attribute as the number scale has no bearing on its meaning. Below are the definitions for each of the shape scores as defined by the dataset.

1) Round

2) Oval

3) Lobular

4) Irregular

When performing experiments in WEKA, the values in this attribute will be replaced with text values to ensure its processing as a nominal attribute.

3.3.2 Distribution

As shown in figure 10, the shape attribute is non-normal and is skewed to the right.

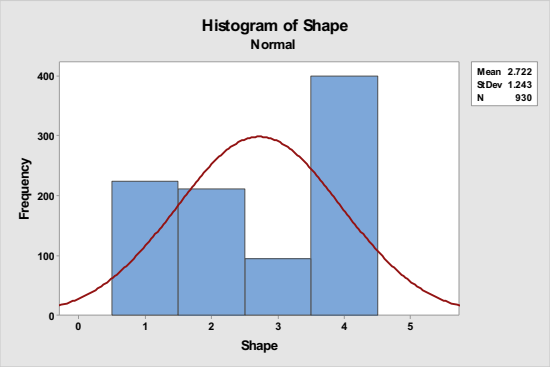


Figure 10: Shape histogram

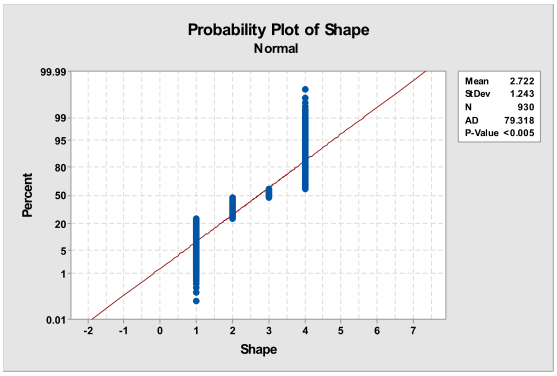


Figure 11: Shape probability

3.3.3 Outliers

While there do not appear to be any outliers within this attribute, it does contain 31 missing values. Due to it being nominal the central tendency for these values should be the mode.

3.3.4 Predictive

In figure 12 below it can be seen that while shape is not on an ordinal scale, a higher score of shape is correlated with a higher chance of a malignant severity.

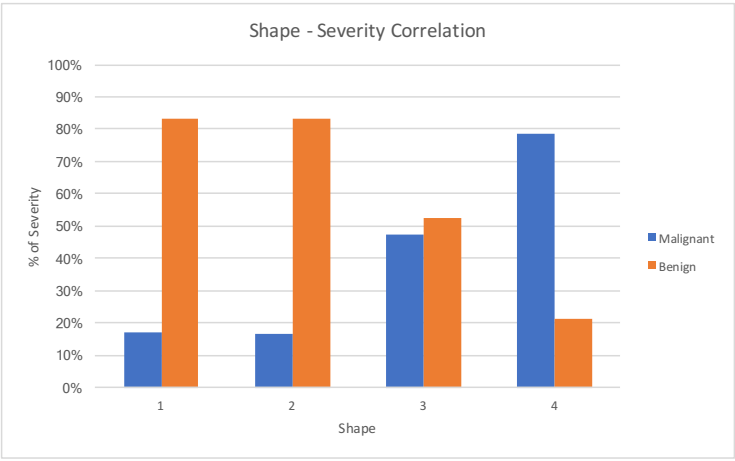


Figure 12: Shape - Severity correlation

3.4 Margin

3.4.1 Measurement

The margin attribute is used to describe the edges of the mass and their distinctiveness from the rest of the breast tissue within the scan. This is also a nominal attribute as there is no scale for the margin descriptors. Below are the 5 different definitions for each of the margin scores.

1) Circumscribed
2) Microlobulated
3) Obscured
4) Ill-defined
5) Spiculated

As with the previous attribute, when performing experiments in WEKA the values in this attribute will be replaced with text values to ensure its processing as a nominal attribute.

3.4.2 Distribution

The distribution of values in the margin attribute appears to be non-normal and is skewed to the right.

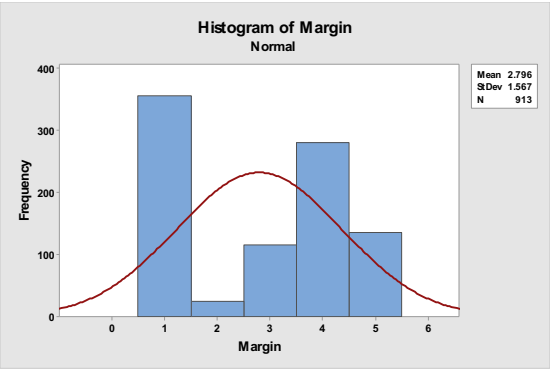


Figure 13: Margin histogram

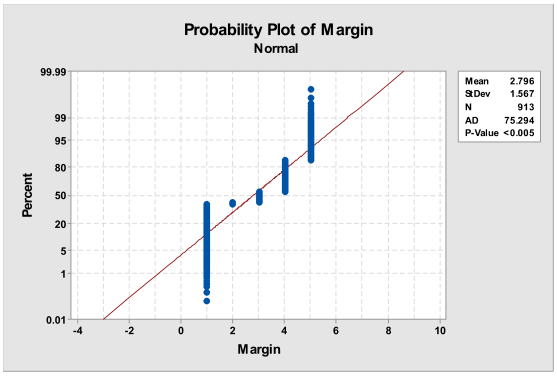


Figure 14: Margin probability

3.4.3 Outliers

There does not appear to be any outliers within this attribute however it does contain 48 missing values. Due to it again being a nominal value the central tendency for these values should be the mode.

3.4.4 Predictive

Once again, it can be observed in figure 15 that a higher score of margin will correlate with a higher chance of a malignant severity.

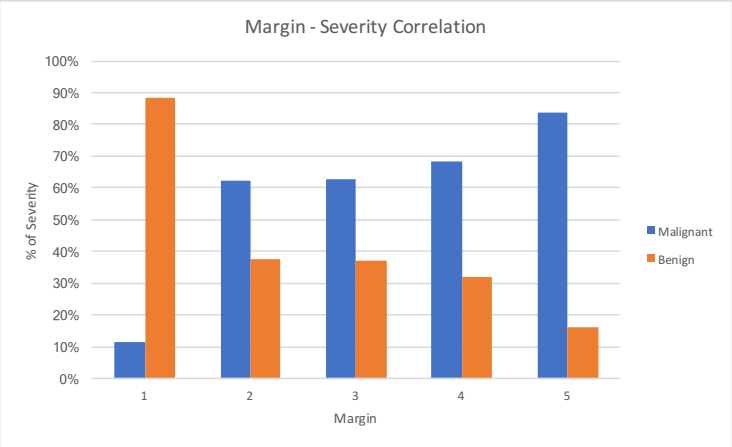


Figure 15: Margin - Severity correlation

3.5 Density

3.5.1 Measurement

This attribute is used to describe the density of the mass on an ordinal scale. A lower score indicates a higher density, which has been shown to be a significant predictor of breast cancer (Woods et al. 2011). Below are the descriptors used to score masses onto the ordinal scale.

1) High

2) ISO (Isodense)

3) Low

4) Fat-containing

3.5.2 Distribution

As shown in figure 13, the distribution of values in the density attribute appear to be non-normal and skewed to the right.

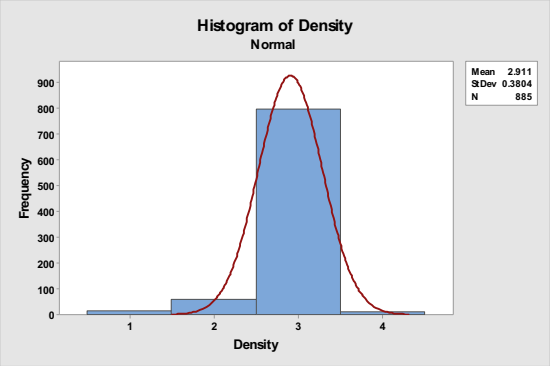


Figure 16: Density histogram

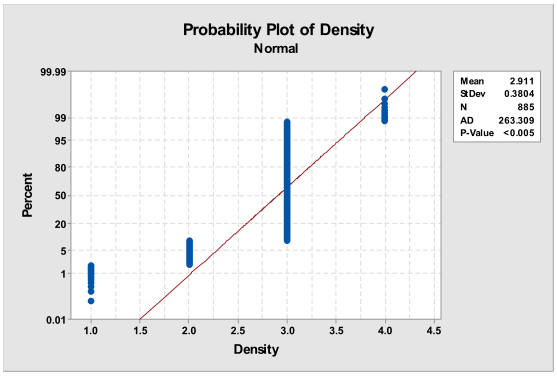


Figure 17: Density probability

3.5.3 Outliers

While there is no outliers within this attribute, there are 76 missing values. The central tendency for replacing these values should be the median due to the data type being ordinal.

3.5.4 Predictive

Figure 18 below shows that in contrary to Woods et al. (2011), the dataset provided does not seem to have a significant correlation between high-density masses and a malignant severity.

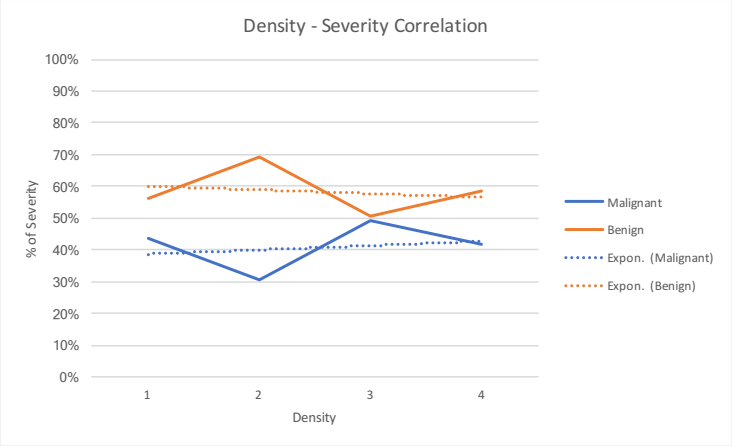


Figure 18: Density - Severity correlation

3.6 Severity

This field is known as a goal field and is a binomial type. This field contains the classification of the instances. In order to make the dataset more compatible with WEKA processing, the original values of '0' and '1' have been replaced by '−' and '+' respectively.

4 Classifier Evaluation

In this section the two different classifier types will be evaluated for their suitability in the task of classifying the mammographic dataset.

4.1 Decision Tree

4.1.1 Strengths

Speed. Decision trees are often a lot faster than neural networks (once trained), this is due to the fact that the algorithm creates a set of simple rules which are very quick to execute, and can be implemented in any programming language.

Interpretability. Again due to the nature of decision trees, the resulting classifier is very easy to interpret due to the creation of simple rules. It then provides insights on the relationships between the attributes and the classes.

Feature selection. Decision trees are good with noisy data due to the fact that they are able to completely ignore certain attributes if they are not deemed important enough as a contributor to classification.

4.1.2 Weaknesses

Nonlinear interactions. Decision trees are not as good at handling data with nonlinear relationships due to their structure. If the relationship between two attributes is not definable in a simple rule then a decision tree will not be able to

4.2 Artificial Neural Network

4.2.1 Strengths

Nonlinear interactions. ANNs are very good at handling abstract relationships in datasets, therefore if there is any information contained in the data which can be inferred, then it is likely that a neural network could make use of this inference.

4.2.2 Weaknesses

No feature Selection. ANNs automatically use every piece of input data given to them. If one or more of the attributes contain noisy data or do not provide much value, then these could hinder the performance of the classifier unless they are removed manually as a pre-processing step.

Black box. ANNs do not provide a systematical view of how a classification is made (it can be done given the proper functions and weight/bias values, however calculating it manually could take a very long time). This is not useful when trying to determine the relationships between different attributes in the data.

Speed. The time taken for both training and classification are often higher in neural networks especially when the complexity of the network structure is increased (i.e. by adding more neurons and/or hidden layers).

4.3 Prediction

In order to predict the outcome of which classifier is most suited for this task, it must be made clear how the dataset will interact with the classifier. In section 3 it was shown that the dataset has some clear links between attributes and classifications (such as a high correlation between age and severity shown in figure 9). Due to the nature of decision trees which are more able to deal with data that have a clear link between attributes and classification, it could be presumed that the decision tree classifier would be more suited for this task than an artificial neural network.

5 Initial Experiments - Pre-processing

In this section the pre-processing steps for the dataset will be explained and a series of initial experiments prior to pre-processing will be carried out to find the best strategy for classifying data.

5.1 Missing values

As explained in section 3, each parameter has its own central tendency due to their different distributions and data types. The strategy for missing values will be to replace them with the central tendency for each parameter. As there is only 961 instances within the dataset it would not be beneficial to remove any instances with missing values as the other parameters would then suffer from real data being removed from them.

5.2 Outliers

As previously explored in section 3.1.3, the main outlier is within the BI-RADS attribute and has been replaced with the central tendency. Once again if this instance were to be removed (when it is only likely there due to human input error) then the dataset could risk losing valuable data for other parameters in that instance.

5.3 Results

Below are the results from some initial experiments done prior to pre-processing of data (aside from removing the outlier mentioned in section 3.1.3).

5.3.1 Decision Tree

(a)

Confidence	Accuracy (%)
0.05	82.2
0.1	82.16
0.15	82.19
0.2	82.27
0.25	82.19
0.3	**82.33**
0.35	82.31
0.4	82.12

(b)

Confidence	Accuracy (%)
0.29	82.29
0.295	82.31
0.299	82.33
0.305	**82.33**
0.31	**82.33**
0.32	**82.33**
0.34	82.31
0.36	82.32
0.38	82.28

Table 1: Tuning confidence

(a)

Min Objects	Accuracy (%)
2	82.33
5	82.30
10	82.24
15	82.54
18	82.83
19	82.99
20	83.00
21	83.04
22	82.98
30	83.16
35	83.53
40	83.73
45	83.80
50	**83.82**
60	83.04
70	82.82

(b)

MinNumObj	Accuracy (%)
45	83.80
46	**83.84**
47	**83.84**
48	83.81
49	83.81
50	83.82
51	83.79
52	83.79
53	83.70
54	83.62
55	83.62

Table 2: Minimum number of objects tuning (C=0.3)

(a)

MinNumObj	Accuracy (%)
46	**83.84**
100	82.50
110	82.50
120	82.23
130	81.89 *
140	81.89 *
150	81.89 *
160	81.89 *
170	81.89 *
180	81.89 *
190	81.89 *
200	81.89 *

(b)

MinNumObj	Accuracy (%)
46	**83.84**
120	82.23
121	82.03 *
122	82.05 *
123	82.03 *
124	81.99 *
125	81.92 *
126	81.88 *
127	81.89 *
128	81.89 *
129	81.89 *
130	81.89 *

Table 3: Finding most pruned tree

5.3.2 Artificial Neural Network

Table 4: Learning rate accuracy from 200-3000 epochs

	Learning Rate				
	0.1	0.3	0.5	0.7	0.9
Epochs	Accuracy (%)				
200	**81.19**	80.72	80.49	80.21	79.93
250	80.19	**80.99**	80.58	80.39	79.97
350	81.27	**81.36**	81.01	80.5	80.2
450	81.49	**81.52**	80.87	80.55	80.39
550	**81.72**	81.56	81.13	81.08	80.74
650	**82.05**	81.7	81.34	81.27	80.92
750	**82.04**	81.68	81.59	81.37	81.14
850	**82.16**	81.77	81.76	81.52	81.16
950	**82.12**	81.9	81.71	81.66	81.32
1050	**82.13**	81.96	81.84	81.77	81.28
1150	81.99	**82**	81.8	81.81	81.34
1500	82.08	**82.22**	81.84	81.92	81.5
2000	82.23	**82.32**	81.88	81.94	81.74
3000	**82.25**	82.24	81.84	81.86	81.69

Table 5: Momentum accuracy from 200-3000 epochs (LR=0.4, HL=A)

	Momentum				
Epochs	0.1	0.3	0.5	0.7	0.9
200	80.59	**80.66**	80.24	79.92	79.31
300	**81.07**	80.96	80.55	80.46	79.34
400	**81.31**	81.17	80.68	80.55	79.49
500	81.33	**81.36**	80.96	80.72	79.39
600	81.45	**81.51**	81.22	80.98	79.5
700	81.55	**81.64**	81.38	81.15	79.48
800	81.52	**81.71**	81.52	81.41	79.52
900	81.69	**81.72**	81.53	81.46	79.38
1000	**81.85**	81.78	81.55	81.42	79.46
1100	81.89	**81.98**	81.61	81.63	79.66
1500	**82.07**	82.04	81.69	81.61	79.9
2000	**82.06**	82.03	81.78	81.66	80.05
3000	81.98	**82.05**	81.81	81.64	80

Table 6: Two hidden layer ANN structure (LR=0.4, M=0.2, E=950)

	Second Layer Neurons				
First Layer Neurons	1	2	3	4	5
1	82.34	82.25	82.43	82.64	**82.67**
2	81.78	81.89	82.29	82.06	**82.38**
3	81.02	81.32	81.79	81.96	**82.01**
4	80.66	81.38	80.92	80.99	**81.07**
5	80.55	80.53	**81.29**	81.02	80.7

Table 7: Learning rate impact on accuracy from 250-3000 epochs (M=0.2, HL=1)

	Learning Rate				
Epochs	0.1	0.3	0.5	0.7	0.9
250	81.73	**81.88**	81.85	81.87	81.8
350	**82.27**	82.24	82.21	82.19	82.23
450	**82.58**	82.49	82.39	82.33	82.35
550	82.55	**82.61**	82.58	82.41	82.33
650	**82.73**	82.7	82.72	82.66	82.45
750	82.77	82.8	**82.83**	82.74	82.5
850	82.79	**82.93**	82.88	82.66	82.5
950	82.89	**83**	82.87	82.78	82.55
1050	82.9	**83**	82.89	82.83	82.61
1150	82.89	**83**	82.87	82.88	82.69
1500	82.98	82.99	**83.09**	83	82.87
2000	83.08	83.15	**83.19**	83.14	83
3000	**83.25**	83.19	83.21	83.18	83.01

6 Main Experiments

In this section the experiments will be carried out which indicate the best possible classifier for both Decision trees and Artificial Neural Networks. These results will include pre-processing steps including replacing missing values with the central tendency and replacing nominal values with their text values to ensure they are processed correctly. Throughout all experiments a 10-fold cross-validation train/test set will be used, this will make use of every instance of the data in training and testing which is vital due to the low number of instances in the dataset.

6.1 Decision tree

As a result of the information presented in section 2.1, a series of experiments will be carried out to find the optimal parameters for training when a J48 decision tree algorithm is used on the mamographical dataset. In this section two trees will be laid out, the first being the tree with the highest classification accuracy and the second being the one which is most pruned but not significantly worse than the highest accuracy tree.

6.1.1 Plan

The first series of experiments will be to find the optimal confidence setting, by iterating through different values at gradually smaller intervals. Once the optimal confidence value is found, then the Minimum number of Object parameter will also need to be tuned in order to find its optimal value. Once again this can be done by increasing the value at set increments and reducing the size of the increments around values which show the highest accuracy results.

6.1.2 Execution

Tables 8a, 8b & 8c illustrate the process of finding the optimal confidence parameter value. As seen in table 8c the optimal confidence value (with the minimum number of objects parameter set to its default of 2) is 0.247, which gives an accuracy score of 82.99%. As many of the accuracy values are repeating with the 3 decimal point intervals seen in table 8c, there is no reason to search through smaller intervals.

Table 8: Confidence tuning

(a)

Confidence	Accuracy (%)
0.05	82.26
0.10	82.81
0.15	82.79
0.20	82.80
0.25	**82.99**
0.30	82.92
0.35	82.57
0.40	82.42
0.45	82.30
0.50	82.19
0.55	82.14
0.60	82.14
0.65	82.14
0.70	82.14
0.75	82.14
0.80	82.14
0.85	82.14
0.90	82.14
0.95	82.14
1.00	82.14

(b)

Confidence	Accuracy (%)
0.21	82.88
0.22	82.88
0.23	82.93
0.24	82.97
0.25	**82.99**
0.26	82.93
0.27	82.93
0.28	82.93
0.29	82.89
0.30	82.92
0.31	82.85
0.32	82.85
0.33	82.81
0.34	82.64
0.35	82.57

(c)

Confidence	Accuracy (%)
0.245	82.97
0.246	82.98
0.247	**82.99**
0.248	**82.99**
0.249	**82.99**
0.250	**82.99**
0.251	82.96
0.252	82.93
0.253	82.93
0.254	82.93
0.255	82.93

Tables 9a and 9b show the process of finding the optimal parameter value for the minimum number of objects, with the confidence value set to the optimal 0.247. Specifically table 9b shows that the optimal value is 52 with an accuracy score of 83.95%. As this parameter is an integer scale there is no way to process smaller intervals.

Table 9: Minumum Number of Objects tuning

(a)

Min Objects	Accuracy (%)
2	82.99
5	82.99
10	83.05
15	83.56
20	83.67
25	83.71
30	83.60
35	83.75
40	83.75
45	83.83
50	**83.90**
55	83.07
60	81.79
65	81.79
70	81.78
75	81.83
80	81.83
85	81.83
90	81.83
95	81.83
100	81.83

(b)

Min Objects	Accuracy (%)
45	83.83
46	83.80
47	83.86
48	83.91
49	83.90
50	83.90
51	83.94
52	**83.95**
53	83.85
54	83.66
55	83.07

Tables 10a and 10b show the process of finding the most pruned tree which is not significantly worse than the tree with the highest classification accuracy which was found in the previous paragraph. Specifically table 10b shows that the most pruned tree which is not significantly worse is one with 56 minimum number of objects. Again this parameter is an integer scale so there is no way to find smaller intervals.

Table 10: Finding the most pruned tree

(a)

Min Objects	Accuracy (%)
52	**83.95**
50	83.90
60	81.79 *
70	81.78 *
80	81.83 *
90	81.83 *
100	81.83 *
110	81.77 *
120	81.47 *
130	81.49 *
140	81.49 *
150	81.49 *

(b)

Min Objects	Accuracy (%)
52	**83.95**
50	83.90
51	83.94
53	83.85
54	83.66
55	83.07
56	82.47
57	82.08 *
58	81.83 *
59	81.83 *
60	81.79 *

6.2 Artificial Neural Network

As a result of the information presented in section 2.2, a series of experiments will be carried out to find the optimal parameters for training when an Artificial neural network is used on the mamographical dataset. In this section two network structures will be laid out, the first being the one with a single hidden layer of neurons and the second being one with two hidden layers of neurons.

6.2.1 Plan

In order to find the best structure for an ANN with both 1 and 2 hidden layers of neurons, first the optimal learning parameters must be found. To do this a series of experiments must be run with different learning rate and momentum values, each over a long series of training times. At first intervals of 0.2 will be used between learning rate / momentum values as the more optimal value can be determined once the best training time is found.

Once the optimal training parameters are found another series of experiments with

different combinations of ANN structures will be carried out, which make use of the training parameters found in the previous experiments.

6.2.2 Execution

Table 11a shows that the best learning rate over the majority of training times is 0.1, and that a training time of 700 epochs is the most optimal time with an accuracy of 82.63%. Using this as a basis, table 11b shows a wider scope of learning rates where it can be observed that 0.1 is still the most optimal value. Table 11c shows finer intervals around the 0.1 value and again highlights that it is still the best value. Finally, table 11d shows the two neighbour values which still give lower scores than the 0.1 value. Therefore it is proven that 0.1 is the optimal learning rate value.

Table 11: Learning rate tuning

(a)

	Learning Rate				
	0.1	0.3	0.5	0.7	0.9
Epochs	Accuracy (%)				
100	**82.3**	82.29	82.13	81.84	81.79
200	82.31	**82.33**	82.17	81.87	81.85
300	**82.47**	82.39	82.04	81.9	81.63
400	**82.53**	82.37	81.93	81.8	81.54
500	**82.54**	82.36	81.88	81.75	81.56
600	**82.6**	82.27	81.86	81.72	81.54
700	**82.63**	82.21	81.83	81.68	81.58
800	**82.62**	82.18	81.83	81.61	81.5
900	**82.62**	82.15	81.8	81.62	81.45
1000	**82.6**	82.12	81.79	81.6	81.37
1100	**82.54**	82.1	81.78	81.59	81.36
1200	**82.55**	82.08	81.77	81.56	81.31
1300	**82.53**	82.06	81.73	81.52	81.28
1400	**82.53**	82.01	81.72	81.5	81.27
1500	**82.52**	82.01	81.73	81.46	81.25
1600	**82.53**	81.96	81.71	81.43	81.23
1700	**82.51**	81.89	81.73	81.4	81.24
1800	**82.51**	81.92	81.73	81.44	81.26
1900	**82.51**	81.9	81.76	81.44	81.27
2000	**82.5**	81.93	81.78	81.46	81.22

(b) With training time at 700

Learning Rate	Accuracy (%)
0.1	**82.63**
0.2	82.51
0.3	82.21
0.4	81.99
0.5	81.83
0.6	81.72
0.7	81.68
0.8	81.62
0.9	81.58
1.0	81.73

(c) With training time at 700

Learning Rate	Accuracy (%)
0.02	82.4
0.04	82.38
0.06	82.56
0.08	82.58
0.1	**82.63**
0.12	82.59
0.14	82.54
0.16	82.47
0.18	82.52

(d) With training time at 700

Learning Rate	Accuracy (%)
0.09	82.6
0.1	**82.63**
0.11	82.59

Table 12a shows a similar experiment to that in table 11a, however this time with momentum as the variable parameter and with learning rate set to its optimal value of 0.1. It can be seen that 0.1 is the most common optimal value over the majority of training times. Furthermore it can be seen that a training time of 900 epochs yields the highest classification accuracy of 82.61% with a momentum value of 0.1. Using this as a basis some further experiments were done with smaller training time intervals, with 920 being the standout optimal training time.

The next two tables (12b & 12c) show the discovery of the most optimal momentum value when the training time is set to 920 epochs. The optimal value which can be seen in these results is 0.17 with an accuracy score of 82.65%. As the accuracy results in table 12c appear to repeat at this interval size, it can be concluded that no smaller intervals need be investigated.

Table 12: Momentum tuning

(a)

	Momentum				
	0.1	0.3	0.5	0.7	0.9
Epochs	Accuracy (%)				
100	**82.44**	82.34	82.30	82.34	81.80
200	82.34	82.35	**82.45**	82.33	81.75
300	82.42	82.45	**82.56**	82.33	81.80
400	82.55	82.49	**82.60**	82.30	81.75
500	**82.58**	82.54	82.50	82.30	81.75
600	**82.58**	82.56	82.47	82.26	81.72
700	**82.60**	82.57	82.43	82.19	81.64
800	**82.58**	82.53	82.46	82.16	81.65
900	**82.61**	82.55	82.46	82.11	81.52
1000	**82.61**	82.53	82.43	82.07	81.48
1100	**82.58**	82.55	82.4	82.05	81.46
1200	**82.57**	82.52	82.36	82.01	81.45
1300	**82.56**	82.55	82.31	81.98	81.45
1400	**82.55**	82.5	82.29	81.96	81.37
1500	**82.58**	82.48	82.28	81.94	81.37
1600	**82.55**	82.49	82.22	81.94	81.37
1700	**82.52**	82.47	82.18	81.95	81.38
1800	**82.51**	82.46	82.19	81.94	81.33
1900	**82.49**	82.44	82.17	81.93	81.31
2000	**82.48**	82.4	82.18	81.92	81.31

(b) With training time at 920

Momentum	Accuracy (%)
0.1	**82.62**
0.2	82.60
0.3	82.55
0.4	82.44
0.5	82.45
0.6	82.38
0.7	82.08
0.8	82.02
0.9	81.52
1.0	52.62 *

(c) With training time at 920

Momentum	Accuracy (%)
0.01	82.57
0.03	82.57
0.05	82.61
0.07	82.61
0.09	82.62
0.11	82.61
0.13	82.62
0.15	82.64
0.17	**82.65**
0.19	**82.65**

The final experiments to be completed are those which identify the best structure for the ANN. Table 13 shows that the best structure for 1 hidden layer is one with 3 neurons, which gives a classification accuracy of 82.65%. As the results do not improve significantly above 10 neurons there are no further experiments for 1 hidden layer. Table 14 shows that the best structure for 2 hidden layers is 8 neurons in the first layer and 10 in the second, once again as the results do not improve significantly there are no further experiments past 10 neurons in each hidden layer, considering the time needed for running experiments with higher values.

Table 13: ANN Structure: One hidden layer

Layer 1	Accuracy (%)
1	81.83
2	81.98
3	**82.65**
4	81.80
5	82.10
6	82.25
7	82.05
8	82.10
9	81.88
10	81.95

Table 14: ANN Structure: Two hidden layers

	Layer 2									
	1	2	3	4	5	6	7	8	9	10
Layer 1	Accuracy (%)									
1	81.73	81.76	81.74	81.78	81.78	81.82	81.88	81.88	81.87	81.87
2	82.14	82.00	81.84	81.88	81.84	81.82	81.99	81.85	81.86	81.57
3	82.12	82.10	82.24	82.20	82.58	81.91	82.12	81.91	82.17	81.78
4	82.37	82.56	82.46	82.57	82.46	82.50	82.77	82.48	82.44	82.54
5	82.36	82.63	82.61	82.62	82.79	82.72	82.71	82.64	82.56	82.80
6	82.08	81.84	82.46	82.64	82.64	82.54	82.61	82.55	82.51	82.72
7	82.08	82.45	81.89	82.58	82.61	82.85	82.73	82.75	82.53	82.84
8	82.28	81.78	82.50	82.25	82.38	82.57	82.60	82.45	82.71	**82.86**
9	82.13	82.15	82.01	82.56	82.24	82.68	82.62	82.73	82.71	82.84
10	82.26	82.27	82.37	82.16	82.64	82.39	82.41	82.65	82.67	82.70

7 Conclusions

7.1 Best Decision Tree

As discovered in section 6.1.2, the best pruned decision tree is the same as the one with the highest classification accuracy. This tree has 7 leaves, with a total of 10 nodes. It is made with a confidence value of 0.247 and a minimum number of objects value of 52. The classification accuracy it provides is 83.95%. Below, figure 19 shows the topographical structure of the tree and its node configurations.

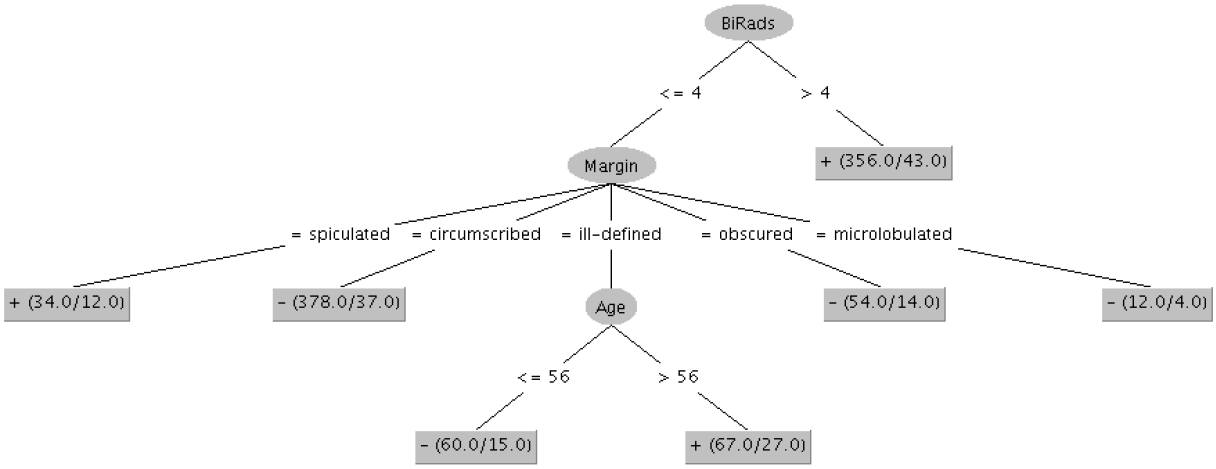


Figure 19: Best pruned & highest accuracy decision tree

7.2 Best Artificial Neural Network

As discovered in section 6.2.2 the best structure for an ANN when applied to this dataset is one with two hidden layers. The first layer has 8 neurons and the second has 10, and the optimal learning rate & momentum are 0.1 and 0.19 respectively with the training time set to 920. The topographical view of the ANN structure described which gives a classification accuracy of 82.86% can be seen in figure 20.

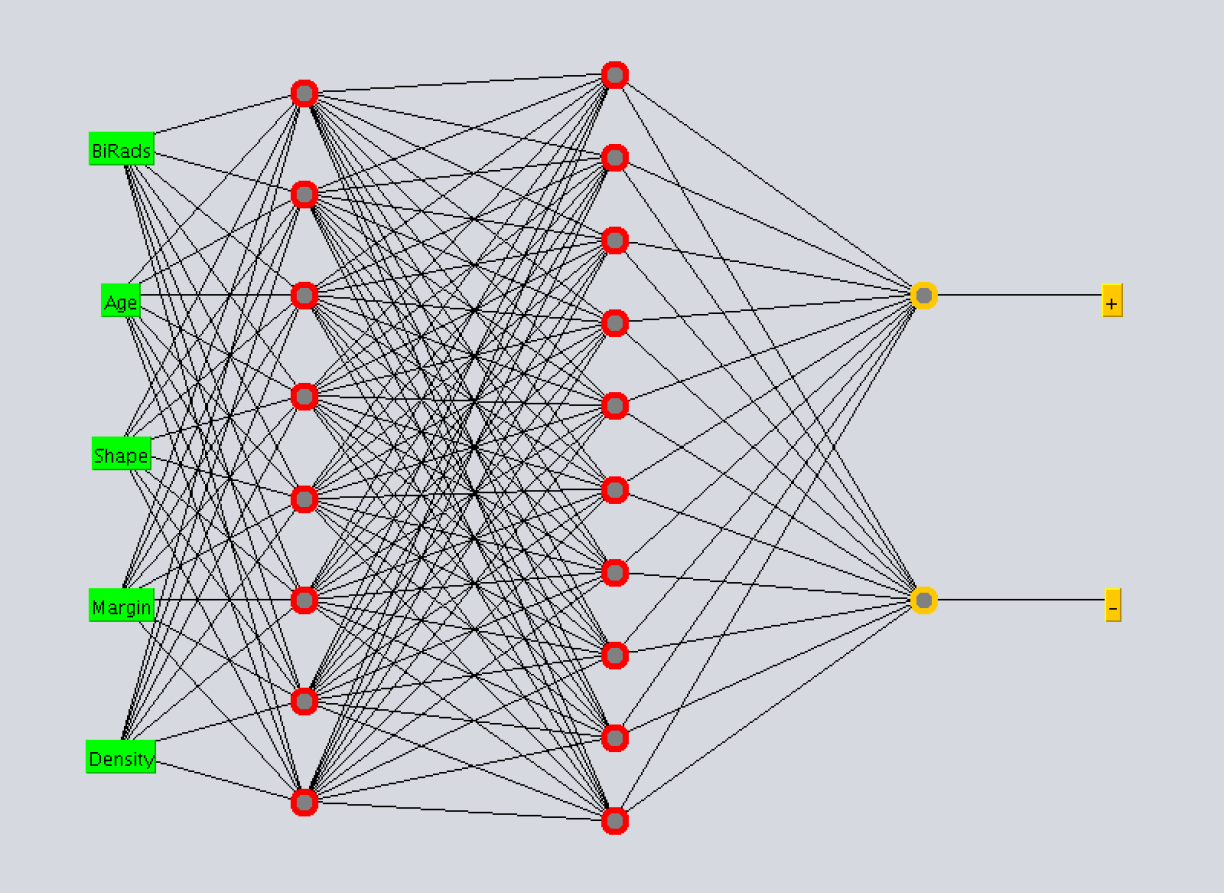


Figure 20: Optimal ANN structure

7.3 Classifier recommendation

The classifier which provided the highest classification accuracy is the J.48 decision tree (see section 6.1.2), which gave an accuracy score of 83.95%. Furthermore the decision tree takes much less time to both train and classify therefore it is highly recommended that the client uses the decision tree for the task of classifying mamographical data.

7.3.1 Discussion

The results displayed in this report highlight that pre-processing steps can increase classification accuracy, for example the highest decision tree accuracy went from 83.84% to 83.95%. The Artificial Neural Network did not perform as well for this dataset, even with complex structures. This could be due to the data not being complex enough for an ANN

to interpret numerical patterns in the data. On the other hand the decision tree was able to pick out much more simple relationships between attributes with simple tests between them.

References

American College of Radiology (1998), *Breast imaging reporting and data system*, American College of Radiology.

Beck, J. R., Garcia, M., Zhong, M., Georgiopoulos, M. & Anagnostopoulos, G. C. (2008), A backward adjusting strategy and optimization of the c4. 5 parameters to improve c4. 5's performance.

D'orsi, C., Bassett, L., Berg, W., Feig, S., Jackson, V., Kopans, D. et al. (2003), 'Breast imaging reporting and data system: Acr bi-rads-mammography', *American College of Radiology* **4**.

Kerlikowske, K., Grady, M. & Barclay, M. (1993), 'Mammography by age', *Jama* **270**, 2444–2450.

Minsky, M., Papert, S. A. & Bottou, L. (2017), *Perceptrons: An introduction to computational geometry*, MIT press.

Quinlan, J. R. (1987), 'Simplifying decision trees', *International journal of man-machine studies* **27**(3), 221–234.

Quinlan, J. R. (2014), *C4. 5: programs for machine learning*, Elsevier.

Safavian, S. R. & Landgrebe, D. (1991), 'A survey of decision tree classifier methodology', *IEEE transactions on systems, man, and cybernetics* **21**(3), 660–674.

Woods, R. W., Sisney, G. S., Salkowski, L. R., Shinki, K., Lin, Y. & Burnside, E. S. (2011), 'The mammographic density of a mass is a significant predictor of breast cancer', *Radiology* **258**(2), 417–425.