Faculty of Computer Science and Engineering

De Montfort University

Leicester, UK

Classification of Skin Tumours through the Analysis of Unconstrained Images

by

Joaquim Mesquita da Cunha Viana – 03071846

jcviana@netcabo.pt

In partial fulfilment of PhD requirements

October/2008

Supervisors:

Dr. John Cowell

jcowell@dmu.ac.uk

Dr. Robert John

rij@dmu.ac.uk

Dr. Mario Gongora

mgongora@dmu.ac.uk

Intentionally left blank

To Ana, Mafalda & Pedro;
my wife & kids
who always believed in me.

Intentionally left blank

List of figures

Figure 2-1 - Image Processing Chain 10
Figure 3-1 - Imaging data collection main window 36
Figure 3-2 - "Non-White" skin Hue, Saturation and Brightness 38
Figure 3-3 - "White" skin Hue, Saturation and Brightness 38
Figure 3-4 - "Non- White" skin Red, Green and Blue 41
Figure 3-5 - "White" skin Red, Green and Blue 41
Figure 3-6 - Melanoma 42
Figure 3-7 - Maximum, Average, NTSC and Intel greyscale images 43
Figure 3-8 - Main edge detection program's panel 44
Figure 3-9 - Sobel filter with 0.5%, 1% and 2.5% low threshold 46
Figure 3-10 - Sobel filter with 5%, 10% and 20% low threshold 46
Figure 3-11 - Marr-Hildreth edges detected with sigma=4 and spreads between 0.1 and 0.5 49
Figure 3-12 - Marr-Hildreth edges detected with sigma=6 and spreads between 0.1 and 0.5 49
Figure 3-13 - Marr-Hildreth edges detected with sigma=8 and spreads between 0.1 and 0.5 49
Figure 3-14 - Marr-Hildreth edges detected with spread=0.1 and sigmas between 4 and 8 49
Figure 3-15 - Canny edges detected with sigma = 4, 5, 6 7 and 8 56
Figure 3-16 - Colour gradient edge detection with thresholds = 10, 20, 30, 40 and 50 57
Figure 3-17 - Hue-Saturation areas with Hue_min = -5, Hue_Max = 70, Sat_min = 0.1 and Sat_Max = 0.6, 0.7, 0.8, 0.9 and 1.0 57
Figure 3-18 - Hue-Sat Histogram areas with Hue threshold = 70% of the maximum Hue histogram value and Sat = 40%, 50%, 60%, 70% and 80% of the maximum Saturation histogram value 57
Figure 3-19- CIELAB filtered image 60
Figure 3-20 - Smoothed image 61
Figure 3-21 - Image resulting from applying the thresholds T_1 and T_2 to the smoothed image .. 62
Figure 3-22 - Output of the Sobel filter applied to the smoothed grey level image 62
Figure 3-23 - Edge resulting from the combination of the outputs 63

Figure 3-24 - Output from applying the *T* threshold 63
Figure 3-25 - Thinned - not yet closed - edge 63
Figure 3-26 - Non allowable pixel configurations 64
Figure 3-27 - Thinner edge (still not closed) 64
Figure 3-28 - Detected edge 65
Figure 3-29 – Asymmetry - Colour program's panel 66
Figure 8-1 - Regions R1 and R2 formed by the Bayesian Classifier for two equiprobable classes 112
Figure 8-2 - Naïve Bayes Network 118
Figure 8-3 - Tree Augmented Bayes Network 119
Figure 8-4 - Support Vectors (Separable Classes) 125
Figure 8-5 - Support Vectors (Nonseparable Classes) 126
Figure 8-6 - Synaptic link 129
Figure 8-7 - Activation link 130
Figure 8-8 - Synaptic convergence or fan-in 130
Figure 8-9 - Synaptic divergence or fan-out 130
Figure 8-10 – Perceptron signal-flow 133
Figure 8-11 - Multilayer Perceptron 134
Figure 8-12 - Signal-flow graph for a single-loop feedback system 135
Figure 8-13 - Hopfield Networks (General and Architectural views) 137
Figure 8-14 - A Hopfield network with 3 neurons 138
Figure 8-15 - A Hopfield network seen as a Perceptron 139
Figure 8-16 - Supervised learning 140
Figure 8-17 - Reinforcement learning 141
Figure 8-18 - Unsupervised learning 142

List of tables

Table 3-1 - Values for Hue, Saturation and Brightness of “White” skin 37
Table 3-2 - Values for Hue, Saturation and Brightness of “Non-White” skin 39
Table 3-3 - Values for Red, Green and Blue of “Non-White” skin 40
Table 3-4 - Values for Red, Green and Blue of “White” skin 40
Table 4-1 - Confusion matrix for naïve Bayes 70
Table 4-2 - Information Score matrix 71
Table 4-3 - Confusion matrix for Fuzzy k-Nearest Neighbour method 75
Table 4-4 - Confusion matrices for a training set of 50% of the total samples 76
Table 4-5 - Confusion matrices for a training set of 60% of the total samples 77
Table 4-6 - Confusion matrices for a training set of 60% of the total samples 78
Table 4-7 - Confusion matrices for a training set of 80% of the total samples 79
Table 4-8 – Fuzzy k-Nearest Neighbour 80
Table 4-9 - Other results for a training set with 50% of the total samples 81
Table 4-10 - Other results for a training set with 60% of the total samples 82
Table 4-11 - Other results for a training set with 70% of the total samples 83
Table 4-12 - Other results for a training set with 80% of the total samples 84
Table 5-1 - Best performer algorithm with 50% samples training set 88
Table 5-2 - TAN performance with 80% samples training set 88
Table 5-3 - Multilayer Perceptron performance with 80% samples training set 88

Abstract

Skin cancer is the most frequent malignant neoplasm for Caucasian individuals. According to the Skin Cancer Foundation, the incidence of melanoma, the most malignant of skin tumours, and resultant mortality, have increased exponentially during the past 30 years, and continues to grow. [1]. Although often intractable in advanced stages, skin cancer in general and melanoma in particular, if detected in an early stage, can achieve cure ratios of over 95% [1,55].

Early screening of the lesions is, therefore, crucial, if a cure is to be achieved.

Most skin lesions classification systems rely on a human expert supported dermatoscopy, which is an enhanced and zoomed photograph of the lesion zone. Nevertheless and although contrary claims exist, as far as is known by the author, classification results are currently rather inaccurate and need to be verified through a laboratory analysis of a piece of the lesion's tissue.

The aim of this research was to design and implement a system that was able to automatically classify skin spots as inoffensive or dangerous, with a small margin of error; if possible, with higher accuracy than the results achieved normally by a human expert and certainly better than any existing automatic system.

The system described in this thesis meets these criteria. It is able to capture an unconstrained image of the affected skin area and extract a set of relevant features that may lead to, and be representative of, the four main classification characteristics of skin lesions: Asymmetry; Border; Colour; and Diameter.

These relevant features are then evaluated either through a Bayesian statistical process - both a simple k-Nearest Neighbour as well as a Fuzzy k-Nearest Neighbour classifier - a Support

Vector Machine and an Artificial Neural Network in order to classify the skin spot as either being a Melanoma or not.

The characteristics selected and used through all this work are, to the author's knowledge, combined in an innovative manner. Rather than simply selecting absolute values from the images characteristics, those numbers were combined into ratios, providing a much greater independence from environment conditions during the process of image capture.

Along this work, image gathering became one of the most challenging activities. In fact several of the initially potential sources failed and so, the author had to use all the pictures he could find, namely on the Internet. This limited the test set to 136 images, only. Nevertheless, the process results were excellent.

The algorithms developed were implemented into a fully working system which was extensively tested. It gives a correct classification of between 76% and 92% – depending on the percentage of pictures used to train the system. In particular, the system gave no false negatives. This is crucial, since a system which gave false negatives may deter a patient from seeking further treatment with a disastrous outcome. These results are achieved by detecting precise edges for every lesion image, extracting features considered relevant according to the giving different weights to the various extracted features and submitting these values to six classification algorithms – k-Nearest Neighbour, Fuzzy k-Nearest Neighbour, Naïve Bayes, Tree Augmented Naïve Bayes, Support Vector Machine and Multilayer Perceptron - in order to determine the most reliable combined process. Training was carried out in a supervised way – all the lesions were previously classified by an expert on the field before being subject to the scrutiny of the system.

The author is convinced that the work presented on this PhD thesis is a valid contribution to the field of skin cancer diagnostics. Albeit its scope is limited – one lesion per image – the results achieved by this arrangement of segmentation, feature extraction and classification algorithms showed this is the right path to achieving a reliable early screening system. If and when, to all these data, values for age, gender and evolution might be used as classification features, the

results will, no doubt, become even more accurate, allowing for an improvement in the survival rates of skin cancer patients.

Acknowledgements

I would like to thank my Supervisors, Dr. John Cowell, PhD., Prof. Robert John, PhD and Dr. M. Gongora, PhD., for all the support they gave me. They have always been there when I needed a helping hand.

I also want to thank Drs. Campos Lopes, M.D, Dr. Pedro Torres, M.D. and my friends Dr. Daniel de Matos, M.D. and his wife Madalena for all the trouble that I gave them, capturing and classifying images of skin lesions.

A special thanks to my colleagues Paulo Pinto and Isabel Alvarez who had the patience to listen to me - during lunches and coffee breaks - and discuss with me when I reached a crossroad.

Last but, not at all the least, to my wife Ana, for all the strength she gave me when I was nearly giving up. For all the time I should have spent with her and – instead - was spent researching…

Contents

1 Introduction 1

1.1 Significance 1

1.2 Work on the field 2

1.3 Objectives 6

1.4 Proposed system 6

1.5 Organization 8

2 Research context 10

2.1 General image classification tasks and algorithms 10

2.1.1 General considerations 11

2.1.2 Pre-processing 11

2.1.3 Data reduction 14

2.1.4 Segmentation 18

2.2 Object recognition 29

3 Research process 32

3.1 Gathering of images 34

3.2 Feature extraction 35

3.2.1 "Normal" skin 35

3.2.2 Initial conversion to greyscale 43

3.2.3 Edge detection 44

3.2.4 Region detection 57

3.2.5 Combined method 58

3.2.6 Size 65
3.2.7 Diameter 65
3.2.8 Jaggedness 66
3.2.9 Colour detection 67
3.2.10 Calculated values 67

4 Training and testing the system 69

4.1 Features evaluation 70

4.2 Results 73

4.2.1 Confusion matrices 75
4.2.2 Other results 80

5 Conclusions 85

6 Future work 90

7 References 91

8 Appendixes 102

8.1 Appendix A - Data collected from the whole training set 103

8.2 Appendix B – Minima, Averages and Maxima 107

8.3 Appendix C – Classifiers 111

8.3.1 Bayesian classifiers 111
8.3.2 k Nearest Neighbour (kNN) 121
8.3.3 Fuzzy k-Nearest Neighbour (Fuzzy kNN) 122
8.3.4 Support Vector Machines (SVM) 123
8.3.5 Neural Networks 128

8.4 Appendix D – Image samples with detected edges 143

8.5 Appendix E – Features 154

8.5.1 Extracted 154
8.5.2 Calculated 161
8.5.3 Evaluation 166

8.6 Appendix F – Weka© result reports 179

8.7 Appendix G - Tree Augmented Naïve Bayes (TAN) tree 315

8.8 Appendix H – Process Flowchart 318

1 Introduction

1.1 Significance

Amaro and Santos [140,141] state that skin cancer is the most frequent malignant neoplasm for Caucasian individuals. Its incidence – number of new cases / year / 100,000 persons – has been consistently rising along the past 40 years. Its treatment during initial phases is simple and results in high cure rates – above 90%. On the contrary, during its more advanced stages, treatment becomes complex, expensive and with a much smaller cure probability; the evolution of the disease and the sequels resulting from the treatment can cause great suffering and in the most problematic cases, eventually lead to death. Malignant melanoma, although representing only 7% of the total number of skin cancers is responsible for more than 80% of the casualties attributed to malignant skin neoplasia. Prevention is therefore, highly recommendable.

According to the Skin Cancer Foundation, the incidence and mortality of skin cancer have increased exponentially during the past several decades, and every year the figure mounts [1]. Although often intractable in advanced stages, skin cancer in general and melanoma in particular, if detected in an early stage, can achieve cure ratios of over 95% [1,55]. United States statistics show about 1 million new cases every year [1,56]. From these, during 2007, 59,940 were melanoma, the most serious type of skin cancer. With an incidence of 33,910 men and 26,030 women, it was responsible for the death of 5,220 males and 2,890 females.

The incidence of melanoma rises rapidly in Caucasians after age 20. Fair-skinned individuals exposed to the sun are at higher risk.

As stated by the American Cancer Society (ACS), [1,56] the melanoma mortality rate in the United States increased by 1.7% annually between 1973 and 1990, then decreased or stabilized thereafter. It is widely hypothesized that the changes in this trend in the 1990s might be due to

prevention and/or early detection practices. Still in accordance to ACS and as stated in [57] *"over 90% of melanomas that arise in the skin can be recognized with the naked eye. Very often, there is a prolonged horizontal growth phase during which time the tumour expands centrifugally beneath the epidermis but does not invade the underlying dermis. This horizontal growth phase may provide lead time for early detection. Melanoma is more easily cured if treated before the onset of the vertical growth phase with its metastatic potential"*.

In Portugal, as shown by Amaro [140] the incidence of melanoma rose drastically during the last 40 years. It is 10 times more frequent today than in the beggining of the sixties. Its incidence more than doubled in the last 15 years and the tendency is to keep on rising. In fact Associação Portuguesa de Cancro Cutâneo points to an incidence of more than 8 / year / 100,000 individuals. These values change widely throughout the world, from around 90 cases / year / 100,000 persons in Australia to 0.5 cases in Japan and India.

Up to now, the analysis and classification of skin spots have been the result of a human expert assessment, with no more than 60% accurate results [120] and (or) of an intrusive biopsy with posterior microscopic analysis of the cells.

1.2 Work on the field

Conscious of the situation, several organizations have dedicated time and efforts trying to improve the early screening process, developing new pieces of hardware and software.

One of the most impressive solutions already developed was the result of Australian Commonwealth Scientific and Industrial Research Organisation research. It is now a part of the commercial solution named SolarScan® and sold by Polartechnics. According to Polartechnics, "*SolarScan® captures and analyses calibrated images of pigmented skin lesions that are stored for subsequent lesion monitoring or confirmation.*
The SolarScan® camera has an integrated light source and employs a three CCD video camera for enhanced colour resolution. Each image is calibrated to a common reference standard to ensure image reproducibility".

The system was tested and Menzies et al published one paper [47] - The Performance of SolarScan – where is stated that "*the melanocytic-only diagnostic model was highly reproducible in the test set and gave a sensitivity of 91% (95% confidence interval [CI], 86%-96%) and specificity of 68% (95% CI, 64%-72%) for melanoma. SolarScan had comparable or superior sensitivity and specificity (85% vs 65%) compared with those of experts (90% vs 59%), dermatologists (81% vs 60%), trainees (85% vs 36%), and general practitioners (62% vs 63%). The intraclass correlation coefficient of intrainstrument repeatability was 0.86 (95% CI, 0.83-0.88), indicating an excellent repeatability*".

American researchers have also been working on a project similar to the one that is described here [48]. There are no publications of any relevant development up to the present moment.

On May, 1st 2006, Cancer Weekly magazine, wrote that an American company, "PhotoMedex, Inc. (PHMD) announced that it has signed agreements with AzurTec, Inc. to resume development and to undertake the manufacture and distribution of AzurTec's MetaSpex Laboratory System (MLS), a light-based system designed to detect certain cancers of the skin".

Walls, Tehrani, Cotton, Moncrieff, and Hall, of Astron Clinica, a Cambridge-based (UK) company, presented a poster under the title "The Non-Contact SIAscope in the Diagnosis of Cutaneous Lesions" [49] during the March, 2006 American Association of Dermatology Meeting. The basic idea presented was an Astron's skin-imaging technology, SIAscopy™ (Spectrophotometric Intracutaneous Analysis). It "*uses visible and infra-red light to examine the main skin components (blood, melanin, dermal melanin and collagen) to a depth of 2mm below the skin's surface, and displays the images on standard PCs. The technology is based on research started at Birmingham University by the company's Chief Scientific Officer, Dr Symon Cotton, and has been developed by Astron in association with the University and Addenbrooke's Hospital, Cambridge. It is now being used worldwide in Astron's DERMETRICS® products suite by dermatologists, plastic surgeons and other medical professionals*".

Nevertheless, most of the apparently available solutions do not deliver data on the reliability levels achieved. In spite of the interesting values claimed by Polartechnics, the idea that was obvious to me through conversations with specialists on the field is: No system, up to this

moment and up to their knowledge, is good enough to be considered reliable. They are, most of all, used as a mere repository of images.

Some academic theoretical work has been published by [143] Hintz-Madsen, M., Hansen, L., Larsen, J., Olesen, E. and Drzewiecki, K., several years ago – 1995 – on the application of Neural Classifiers to skin lesions classification purposes. According to the authors, their "*pruned network classified 74% of the lesions correctly on the training set and 66% on the independent test set. The fully connected network classified 98% correctly on the training set and 66% on the test set*"

On the subject of skin lesions differentiation, She and Fish published a paper [144] where they claim to be able to identify – not classify - skin lesions "using skin line direction".

Sigurdsson, Larsen, Hansen, Philipsen, and Wulf [145] in 2002 published a paper where Bayes rule is used for skin lesions classification with rather inconclusive results.

More recently, during the research process, the author became aware of a paper by Cheng, Y., Swamisai, R., Umbaugh, S, Moss, R., Stoecker, W., Teegala, S. and Srinivasan, S. [142]. They claim *"overall classification success of 79%, with 70% of the benign lesions successfully classified, and 86% of malignant melanoma successfully classified."*

Since all the research on this subject, which is limited in numbers, is mainly to be included in commercial products, the information available is very scarce. It has not been possible to get much more data than those within commercial leaflets and a few isolated technological papers. Nevertheless, this kind of tasks – pattern recognition and in more depth, image processing and classification - have already been the subject of many technical and scientific papers related to their application to other areas of human activity.

In the literature, several other applications for classification algorithms can be found, namely those relying on image processing. Here are some examples for various areas of activity:

- Cocosco, Zijdenbos and Evans [149] published a paper on brain tissue classification through the analysis os MRI images and posterior utilization of a non-parametric algorithm to classify the images;
- In their paper - Matching Shapes [150] - Serge Belongie, Jitendra Malik and Jan Puzicha, present a method for "*measuring similarity between shapes and exploit it for object recognition*". They previously defined various features as basic characteristics of the object to classify and tried to automatically recognize its shape. This was done using k-Nearest Neighbour classifiers;
- Fortson, Lynch, and Newell wrote a paper [151] about "*automatic classification and quantitatively measure the extent of a lung disease called Scleroderma using High Resolution Computed Tomography (HRCT) imagery*". In their work they used standard deviation, skewness and kurtosis of the image intensities within local neighbourhoods whose values are then fed to 17 Maximum likelihood classifiers, in order to detect anomalies in the lung's tissues ;
- Segmentation can also be achieved through image compression. Yang et al. in their 2007 paper [152] stated that natural-image segmentation can be seen as a problem of clustering texture features. The approach they used consists on oversegmenting the images with low level segmentation based on local values for colour and edges, into several hundred segments. They then interconnect the segments impose the constraint that two segments Si and Sj can be merged together only if they are spatially adjacent in the 2D image.
- Furthermore, one of the most known applications for image segmentation is optical character recognition. In the book Character Recognition Systems: A Guide for Students and Practicioners [153] various algorithms are described, both for feature extraction and for pattern classification, as well as several methods for word and string recognition.

1.3 Objectives

The aim of this research was to evaluate the possibility of development and eventual design and posterior implementation of a system that, under present technological conditions was able to classify skin spots as inoffensive or dangerous, automatically and within an acceptably small error margin; if possible, smaller than the results achieved by a human expert and every system already developed.

1.4 Proposed system

The proposed system uses a novel approach to classify skin lesions. It does so by capturing an unconstrained image of the affected skin area, extracting its characteristic features and classifying the skin spot as a member of one of two classes: Melanoma and Other. This is achieved by using a Bayesian statistical process - both a plain Naïve Bayes algorithm and a Tree Augmented Naïve Bayes method – a simple k-Nearest Neighbour classifier and a Fuzzy k-Nearest Neighbour, a Support Vector Machine and an Artificial Neural Network.

Some of the major novelties in this research are related to the kind of features that were considered relevant and eventually used for the classification process. All this feature extraction work was strongly dependent on an accurate and fast enough edge detection method. Although based on several proven algorithms, the method finally selected for this purpose had several changes implemented, namely, the definition of thresholds, and the closing and thinning of the edge lines.

Since it was intended to work on unconstrained images, the classification features were then chosen - in a totally innovative way – guaranteeing that they are, as much as possible, independent from the environmental conditions during image capture. Except for the values related to the lesion's size, features result either from ratios or differences between intrinsic characteristics of every image.

From the first two paragraphs, it seems obvious that such a system will help speeding up the screening of malignant tumours in early stages and in this way, contribute to a better public health. Furthermore, the storage of information regarding individuals, allows for the automatic follow up of their skin health, through the analysis of the evolution of the recorded patterns, namely, changes in any of the four main classification characteristics [84,85] - Asymmetry, Border, Colour and Diameter - without a mandatory medical specialist intervention. Combined with the eventual development of a device small enough to be portable, this system may become important personal equipment in where early skin cancer prevention and detection is concerned.

On the other hand, if a reliable enough detection and classification method – one that would give us a rate of errors similar to or smaller than what we could get from an expert dermatologist action – can be built, it may become an important pillar for a system that will allow us to collect images from patients skin spots, far from the dermatology centres and have them verified by one of these equipments. Once classified and if necessary, the patients could then, and only then, be redirected to a regional hospital for more thorough examination.

Some other issues, although not particularly related to the lesion itself, are the individuals' age and more or less fair skin. In what concerns the influence of gender, the starting values considered can be those expressed in [58] for the item Skin – 56% for males and 44% for females. Within the scope of this work, since all the gathered images – shown on section 8.4 Appendix D - were anonymous, neither of these characteristics was considered. All the extracted features relate exclusively to the images themselves.

The work can then be described as a chain of three well identified macro stages:

1. An initial segmentation process, consisting of:
    a. An optional initial conversion to greyscale;
    b. The application of an elementary edge detection algorithm and the combination of its results with another intermediate image, resulting from the selection of pixels with values between two dynamically calculated threshold values;
    c. The thinning and closing of the resulting contour;

   d. Its combination with the original image –see results in section 8.4 Appendix D;

2. A feature extraction process, during which, several image characteristics - whose final and intermediate values can be seen on Apendixes A, B and E - are quantified;
3. A classification task where various specialized algorithms are compared as a means to evaluate their performance – see detailed results on section 8.7 Appendix F.

1.5 Organization

Chapter 2 describes the context for this research, namely the various tasks necessary to reach a relevant feature extraction and a subsequent reliable recognition of the lesions. Several classes of image preprocessing as well as some feature extraction and classification classes of algorithms are described. An outlook of other lines of work already being processed by researchers in the field is also provided in this chapter.

Chapter 3 describes the research process followed along this work, specifically, how images were gathered, the various algorithms used to extract its relevant classification features, starting with segmentation and following with the calculation of size, diameter, asymmetry and colour, as well as all the resultant ratios, for every processed image.

Chapter 4 takes all these features and shows how they were used to train the system and test the outcome of its application, pointing out the results achieved by the various algorithms used. Several statistic ratios used to verify the system's performance are also described in this chapter. The measured results are shown in this chapter - under the form of confusion matrices and statistic ratios tables.

Chapter 5 presents the conclusions drawn from the implementation of the whole process, comparing the performance of the various classification algorithms used. This task takes into consideration the size of the training sets and ranks the classifiers according to their accuracy.

Chapter 6 describes the directions for future work, namely, the broadening of the number of lesions per processed image and the possibility of including the sotware in a portable hardware device.

All the references considered by the author as significant for the developed work are shown in Chapter 7.

Within Chapter 8, in Appendix D, is shown the whole set of 136 processed images. Appendixes A and B, respectively, present all the data collected from them; average, maximum and minimum values for Red, Green and Blue, Hue, Saturation and Brightness. A description of all the base classification algorithms used throughout the work is contained in Appendix C, namely: Bayesian, k-Nearest Number, Support Vector Machines and Neural Networks. Appendix E contains the values for all the extracted and calculated features as well as the evaluation of their relevance according to several different criteria. Appendix F includes the Weka classification reports for all the algorithms used for training and testing the system. The Augmented Naïve Bayes tree is drawn in Appendix G and, finally, Appendix H shows a flowchart of the whole process.

2 Research context

2.1 General image classification tasks and algorithms

As can be concluded from the 2002 paper by Egmont-Petersen et al. [9], the problem of Image Recognition has been addressed in a number of papers and even implemented on a series of applications. According to Egmont-Petersen et al. [9], what we call Image Recognition is, in fact, a series of steps integrating what the author calls image processing chain – represented by fig. 2-1.

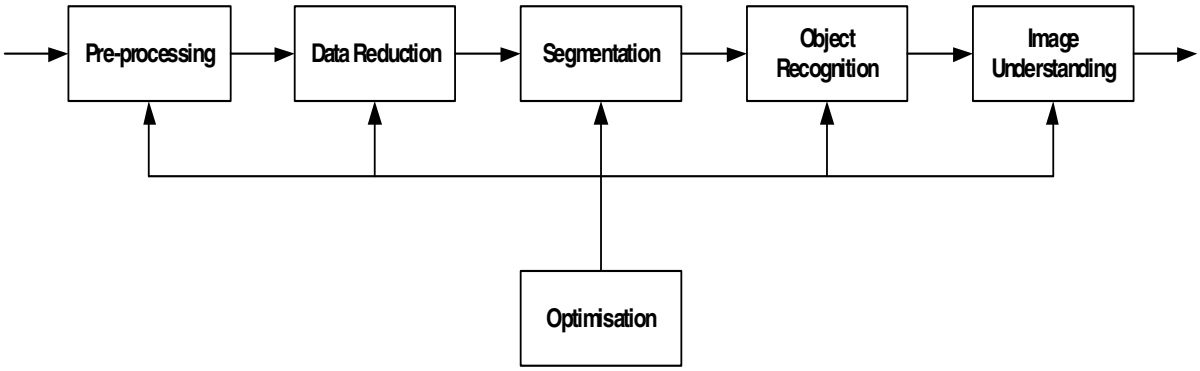


Figure 2-1 - Image Processing Chain

Since the objectives that are being pursued with this project are very specific, within this work, a somewhat simpler process was considered. In fact, for the purpose of this work – classification of skin tumours – only the tasks related to the Pre-processing, Data Reduction, Segmentation and Object Recognition phases of figure 2-1 were considered.

2.1.1 General considerations

Our problem is, clearly, one of classification. According to Bishop [6], in these cases where a wrong classification can lead to a disastrous health situation for the patient, *"the selection criterion should, ideally be taken to be the probability of misclassification or, more generally as the expected total loss or risk"*.

Assuming that, in accordance to all the medical texts that were consulted by the author, all the basic characteristics of a melanoma are those that were underlined by a popular method for remembering the signs and symptoms of melanoma known as "ABCDE" [84,85,95]:

Asymmetrical skin lesion.

Border of the lesion is irregular.

Colour: melanomas usually have multiple colours.

Diameter: moles greater than 5 mm (0,1969 in) are more likely to be melanomas than smaller moles.

Evolution: The evolution (i.e. change) of a mole or lesion may be a hint that the lesion is becoming malignant.

Asymmetry, Border, Colour, Diameter and Evolution – are in fact quite relevant to the hypothetical classification of skin tumours. It became obvious from the conversations with dermatology specialists, that each one by itself is not enough to allow for a near perfect classification of skin tumours. So, this work had to determine if there were other characteristics that might be important for the task to accomplish and, more than this, which were the combined features that might be relevant to this work's main objectives.

2.1.2 Pre-processing

Within the scope of this work, pre-processing was needed for testing some of the possible research paths, particularly to convert the original image to some different colour space. This

problem is normally addressed by applying a combination of well known filter algorithms to the captured image.

2.1.2.1 Von Kries Transform

One of those filters is the Von Kries transform. It is based on colour constancy, a property of the human eye that allows us to identify the colour of objects as more or less constant, independently of the illumination conditions of the moment [134]. According to Johannes von Kries, the human retina has three different colour-sensitive cone types, which are impressed by Long-, Medium-, and Short-wavelength stimuli. This resulted in a colour space – LMS – to which RGB or XYZ [128,129,135] values can be converted by the application of a 3x3 matrix. After this operation, the three LMS primary values are scaled to balance the neutral. The original colour can then be reached by converting the LMS colour space to the original one. The von Kries transform matrix can be described by:

$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} \frac{1}{L'_w} & 0 & 0 \\ 0 & \frac{1}{M'_w} & 0 \\ 0 & 0 & \frac{1}{S'_w} \end{bmatrix} \begin{bmatrix} L' \\ M' \\ S' \end{bmatrix}$$

Where L, M, and S are the colour-balanced LMS values and L'_w, M'_w, and S'_w are the tristimulus values of a white object in the unbalanced colour image, and L', M', and S' are the tristimulus values of a pixel in the unbalanced colour image.

RGB values can be converted to CIE XYZ colour space and afterwards, these pixel values can then be converted to the LMS colour space, using among others, the coefficient matrixes defined by one of the CMCCAT2000 [136], RLAB [138] or CIECAM97s [137].

2.1.2.2 Homomorphic Filter

The application of this filter intends to improve the result of a capture that took place with poor illumination conditions. It includes the following steps:

1. Taking the log of all pixels in the image.
2. Computing the Fourier Transform of the collected image.

   The Fourier transform of a continuous one dimension function is given by:

$$F(w) = \int_{t=0}^{\infty} f(t)e^{-jwt}dt \tag{2-1}$$

   Since we work with pixels – discrete values – it can be written as:

$$F(w) = \sum_{k=0}^{N-1} f(k)e^{-\frac{2\pi jwk}{N}} \tag{2-2}$$

   And in two dimensions:

$$F(x,y) = \frac{1}{\sqrt{nm}} \sum_{i=0}^{n-1} \sum_{k=0}^{m-1} e^{\frac{-2\pi j(xi+yk)}{nm}} f(i,k) \tag{2-3}$$

   This transform is calculated by applying the filter to each row of the original image; which will result in another already semi-filtered image and then applying the same filter to every column of the latter.

   Its inverse form is almost exactly the same, only with the sign reversed.
3. Apply a high emphasis filter to the resulting image;

   Since the Fourier transform will act negatively upon the high frequencies, the level of detail within the transformed image will be reduced. To improve this level of detail, a filter that gradually improves the amplification of the input pixels as their frequencies grow is applied to the Fourier transformed image. This is done by multiplying the filter, pixel-by-pixel, with the Fourier transform of the log of the initial image.

4. Compute the inverse Fourier transform.
5. Compute the exponential of every pixel in the image.

   This will reverse the logarithms applied on step 1, restoring the original, but enhanced image.

2.1.2.3 Scaling

To compensate for differences between the distances from the camera lens to the image being captured it would be necessary to execute a scaling transform to the captured image. This is a very simple operation [133] which consists on multiplying the two image coordinates *x* and *y* by some scaling factor.

The scaling operator with growth *a* along the *x* axis and *b* along the *y* axis can then be described as:

$$T_{a,b}(x, y) = (a \times x, b \times y) \quad (2\text{-}4)$$

As a way to reduce the influence of scaling, most of the features used to classify the lesions within this research, are ratios, rather than absolute values – although not the only one, this is one of the major novelties of this work – images can be collected with different palettes, sizes, illumination, image formats or even resolutions. The tests with basic edge detection algorithms were executed with, at most, simple conversions to greylevel. The last method used and, in fact, the one adopted as a basis for the combined edge detection system, is supported on a previous conversion to CIELab colour space, as can be seen in section 3.2.5.

Since all the images processed during this research were assumed to be captured with the same camera focal distance, the scaling effect introduced by the differences in the camera position was not taken into account.

2.1.3 Data reduction

One of the main problems of image recognition is the so called *"curse of dimensionality"* [3,5,6], which derives from the fact that the number ***N*** of points necessary for an acceptable performance of such a system, grows exponentially with the number ***l*** of dimensions considered, that is, if ***N*** equidistant points are necessary to process one dimension of an object, then, every dimension will need a corresponding number of points to be processed. If the object has the same size along two dimensions, N^2 points will be needed. For three equal sized dimensions, N^3, and so on [3]. In general, the number of features or combinations of features is too large for being analysed by a Neural Network.

Since the training did not have a great number of images, the reduction of features to the least number possible, as a means to also reduce the required training time, without relevant losses in accuracy, was an important issue. With this in mind, every combination of the main four characteristics referred above was assessed so that mutual correlations could be calculated and irrelevant features discarded. Nevertheless, since for the purpose of this work, precision – and not the time needed to process the images - was the most important issue, all the extracted – and considered relevant – features were taken into consideration.

According to Bishop [6], feature selection must rely on two basic components: an adequate selection of relevant features and a systematic procedure for searching through candidate subsets of features. The values that every feature takes for the different classes have to be verified in order to check if they differ significantly – in which case the feature is adequate to this work's purposes - or not.

2.1.3.1 Features Evaluation

Some examples of methods that may be used to drive the selection of relevant features for classification purposes are show here. The described algorithms were used for feature evaluation within the scope of this research. The ranking results can be seen under points 4.1 and 8.5.3 - Appendix E, of this thesis and were calculated with the open source package Weka©.

2.1.3.1.1 Information Gain

This concept intends to represent the expected information and can be seen as the change in information entropy – H – from a situation where the attribute feature had not yet been considered and the state of the system after using it. It is given by:

$$Information\ Gain(Class, Attribute) = H(Class) - H(Class|Attribute) \quad (2\text{-}5)$$

Entropy – $H(X)$ – for a random variable X with n outcomes x_i with $i = 1, 2, \ldots n$, as can be seen in section *2.1.4.2.4* is given by:

$$H(X) = -\sum_{i=1}^{n} p(x_i) log_b p(x_i) \quad (2\text{-}6)$$

All the feature variables used for this research work are binary and so, this expression becomes:

$$H(X) = -\sum_{i=1}^{n} p(x_i) log_2 p(x_i) \quad (2\text{-}7)$$

2.1.3.1.2 Gain Ratio

This evaluation criterion is based on the previous one. It relates the Informaton Gain to the intrinsic entropy of the attribute being evaluated, in fact, normalizing the results in relation to the number of possible outcomes of a given attribute. It is defined as:

$$Gain\ Ratio(Class, Attribute) = \frac{\big(H(Class) - H(Class|Attribute)\big)}{H(Attribute)} \quad (2\text{-}8)$$

2.1.3.1.3 Principal Components Analysis

The process of dimensionality reduction, also known as the Karhunen-Loève transform can be defined [4] as an attempt to map vectors on a *d*-dimensional space – $x_1, x_2, \ldots x_d$ – onto a smaller *m*-dimensional one, with vectors $z_1, z_2, \ldots z_m$.

Since vector **x** can be represented as a set of orthogonal vectors, it is possible to define:

$$\mathbf{x} = \sum_{i=1}^{d} z_i \mathbf{u}_i \quad (2\text{-}9)$$

The vectors are orthonormal, i. e. they satisfy the relation:

$$\mathbf{u}_i^{\mathrm{T}} \mathbf{u}_i = \delta_{ij} \quad (2\text{-}10)$$

Where δ_{ij} is the Kronecker symbol, defined as 1 if $i = j$ or 0 otherwise.

z_i can then be written as:

$$z_i = \mathbf{u}_i^{\mathrm{T}}\mathbf{x} \tag{2-11}$$

And this can be seen as a passage from the coordinate system of the **x** vectors to the one of the **z** vectors. If we keep only a subset $K < d$ of the original vector elements, the coefficients of the remaining elements will be replaced by constants c_i and every **x** vector can approximately be represented by:

$$\tilde{\mathbf{x}} = \sum_{i=1}^{K} z_i \mathbf{u}_i + \sum_{i=K+1}^{d} c_i \mathbf{u}_i \tag{2-12}$$

Our task is to select the c_i coefficients that let us reach the best approximation to the original vectors. Assuming a set of N vectors, $\mathbf{x}^n$, where $n = 1, 2, \ldots, N$, since the error introduced by this process to every original vector is:

$$\mathbf{x}^n - \tilde{\mathbf{x}}^n = \sum_{i=K+1}^{d} (z_i^n - c_i)\mathbf{u}_i \tag{2-13}$$

The best approximation will be the one that reduces to a minimum the square of the errors over the whole set of vectors which is equivalent to minimizing:

$$E_K = \frac{1}{2}\sum_{n=1}^{N}\sum_{i=K+1}^{d} (z_i^n - c_i)^2 \tag{2-14}$$

If we then set the derivative of this expression with respect to c_i to zero, we get:

$$c_i = \frac{1}{N}\sum_{n=1}^{N} z_i^n \tag{2-15}$$

If now we define the mean vector $\bar{\mathbf{x}}$ as:

$$\bar{\mathbf{x}} = \frac{1}{N}\sum_{n=1}^{N}\mathbf{x}^n \tag{2-16}$$

We can write the previous expression as:

$$c_i = \mathbf{u}_i^{\mathrm{T}}\bar{\mathbf{x}} \tag{2-17}$$

And the sum-of-squares error as:

$$E_K = \frac{1}{2}\sum_{i=K+1}^{d}\sum_{n=1}^{N}\left\{\mathbf{u}_i^{\mathrm{T}}(\mathbf{x}^n - \bar{\mathbf{x}})\right\}^2 = \frac{1}{2}\sum_{i=K+1}^{d}\mathbf{u}_i^{\mathrm{T}}\boldsymbol{\Sigma}\mathbf{u}_i \tag{2-18}$$

Where $\boldsymbol{\Sigma}$ is the covariance matrix of the set of vectors and can be seen as:

$$\boldsymbol{\Sigma} = \sum_{n}(\mathbf{x}^n - \bar{\mathbf{x}})(\mathbf{x}^n - \bar{\mathbf{x}})^{\mathrm{T}} \tag{2-19}$$

And the task, now, resumes to minimizing E_K with respect to the vectors $\mathbf{u}_i$, what can be proven to be equivalent to make $\boldsymbol{\Sigma}\mathbf{u}_{\mathrm{i}} = \lambda_{\mathrm{i}}\mathbf{u}_{\mathrm{i}}$, and with this, guarantee that they are eigenvectors of the covariance matrix. The minimum error will then be achieved when the $d - K$ smallest eigenvalues are chosen. Each one of the $\mathbf{u}_i$ vectors is called a Principal Component.

2.1.4 Segmentation

The purpose of this phase is, as its name points out, the partitioning of the image into coherent segments, according to some pre-defined criteria [9]. The segmentation task is executed through the analysis of the individual pixels within the image so that it may be possible to identify either edges or areas with different colours. Programs were developed to detect edges/regions the images, acting either directly upon the original colour image or a greyscale converted image.

2.1.4.1 Segmentation process

The image was then analysed in order to detect edge [112,117] pixels – boundary pixels between two different coloured areas. The main problem with this task is the definition of an adequate threshold value for region segmentation. The selected technique was, of course, chosen according to the results obtained by experimental work, both on greyscale and colour segmentation

Some of the most used segmentation algorithms may be defined as belonging to one of the following classes [4,45,46]:

- Template based;
- Laplacian filter;
- Iterative selection;
- Entropy;
- Minimum error thresholding.

Examples of the results achieved by applying these types of edge detection algorithms can be seen in section 3.2.3, 3.2.4 and 3.2.5.

In 1971, Beucher and Lantuéjoul [148] presented a paper with the base concepts of what they called "contour detection by watersheds". In their paper, and using an analogy to hydraulics, they define greyscale valleys or catchment basins – areas with lower intensity levels – and mountains – where the intensity is at a peak, relative to the neighbour pixels. In such situation, water – greylevel intensity – could, like rain drops, either flow from the top of the mountains to the basins or rise from the basins up. In either case, the water level will stop when the maximum intensity levels are reached. Vincent and Soille, in 1991 [146], introduced a significant improvement for the watershed segmentation algorithm. In fact their method avoids the possible appearance of intensity "plateaus" which could represent rather thick detected edges.

Ruzon and Tomasi [94] also developed a rather interesting edge detection system acting directly upon colour images. Nevertheless, since this research is centred on images with one single lesion and the method here developed gave rather good results - see section 8.4 – Appendix D in this document – it was decided not to use it here. I am sure that it will be very useful for future work concerning the processing of images with more than one lesion.

2.1.4.2 Conversion to grey level

Before applying the greylevel segmentation algorithms, colour images had to be converted to greyscale. For that purpose, several methods were tested.

This task is to be accomplished through the detection of areas' edges. According to Bernd Jähne [4,45,46], *"the task of edge detection requires neighbourhood operators that are sensitive to changes and suppress areas of constant values"*. For this task it was decided to try the processing of grey-level images, as well as the direct treatment of colour images. The first developed action was the conversion between colour values and grey values. This was accomplished by assigning a relation between the colour levels of the image being processed and 256 levels of grey that can be represented by an 8 bits word. The main objective was increasing the contrast between different colours without changing their relation to each other. Five different algorithms were tested in order to verify which one was the most adequate to the task:

- The maximum value from Red, Green and Blue;
- The average $\frac{\text{Red} + \text{Green} + \text{Blue}}{3}$;
- NTSC luminance standard $0.299 \times \text{Red} + 0.587 \times \text{Green} + 0.114 \times \text{Blue}$;
- Intel's image library formula[147]
  $0.212671 \times \text{Red} + 0.71516 \times \text{Green} + 0.072169 \times \text{Blue}$;
- Conversion from the RGB colour space into the CIELab [128] colour space.

The first four algorithms, although straightforward to apply, do not take into account the different spaces between colours that can be found within RGB or even HSV colour spaces. On

the contrary, CIELab uses dimensions that vary linearly and so, conversion from colour to greyscale is much more accurate than with any one of the four previous methods.

Examples of the application of these algorithms on a sample image can be seen on section 3.2.2 and 3.2.5.

2.1.4.3 Segmentation by edge detection

In the next sections, the some of the most paradigmatic of these types of algorithms are described.

2.1.4.3.1 Template based

Since an edge is defined by an abrupt change in grey level, these operators have been created in a way that they will be sensitive to these variations. Being a two dimensional reality, an image must be processed along its main pixel directions; columns – ***x*** – and lines – ***y***. The variations in intensity within the image can be well represented by the derivative in *x* and *y* as components of the actual direction along the axes and, afterwards, computing the vector sum. This is called the intensity gradient (∇) and can be represented by the expression below, if an image is considered a function of two variables $\boldsymbol{I}(x,y)$:

$$\nabla I(x,y) = \left(\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y}\right) \tag{2-20}$$

Because an image is formed by discrete points the derivative at a pixel is approximated by the difference between the intensities of the pixels enclosed within some neighbourhood area. The simplest way to calculate this is by considering the differences in intensity between two adjacent points, along the two main directions:

$$\nabla_{x1} I(x,y) = I(x,y) - I(x-1,y) \tag{2-21}$$

$$\nabla_{y1} I(x,y) = I(x,y) - I(x,y-1) \tag{2-22}$$

Since this approximation calculates the gradient at the point (x - ½, y - ½), shifting the edges by one half of a pixel in both directions, a better approximation would be an operator symmetrical with respect to the pixel at location (x, y), given by the expressions:

$$\nabla_{x2} I(x,y) = I(x+1,y) - I(x-1,y) \tag{2-23}$$

$$\nabla_{y2} I(x,y) = I(x,y+1) - I(x,y-1) \tag{2-24}$$

The resulting vector contains all the information necessary to determine the intensity of the variation, as well as its direction. According to Pythagoras theorem, the gradient's intensity will be given by:

$$G_{mag} = \sqrt{\left(\frac{\partial I}{\partial x}\right)^2 + \left(\frac{\partial I}{\partial y}\right)^2} \tag{2-25}$$

and its direction shall approximately be:

$$G_{dir} = atan\left(\frac{\frac{\partial I}{\partial y}}{\frac{\partial I}{\partial x}}\right) \tag{2-26}$$

The classification of a pixel as an edge depends on its gradient's magnitude being above a predefined threshold. If theoretically this approach is quite interesting, in reality, images are quite subject to noise and so, it is possible to classify as belonging to an edge, a point that, in fact, does not belong to it – a **false positive** – as well as failing to detect one point that does belong to the edge being detected, creating what is called a **false negative**.

To calculate the above values, it is normal to use a reduced template as a model for the influence of neighbour pixels on the approximation to the derivative operator on the processed point.

The Sobel [45,96,97,98] operator which was used in this work – and is described further on – is an example of these fixed template edge detection algorithms. The Canny [73,99,100,101]

Operator, can also be seen as an example of this type of algorithms, although with an optimized convolution template.

This type of operators has a very fast execution time. The Canny operator, much more elaborate than Sobel's gives, accordingly, much better results, although not good enough to be considered a solution by itself.

2.1.4.3.2 Laplacian filter

By definition, "*an edge pixel must be near to the boundary between an object and the background, or between two objects*" [46]. The threshold between two areas is found by computing the Laplacian" [102,103,104,105] of the image. A simple way to do this is to convolve the image with the mask:

$$\begin{matrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{matrix}$$

The image histogram is then calculated, considering only pixels with large Laplacians. The level of accuracy will depend on the Laplacian defined threshold level.

The Marr-Hildreth [72,115,116] algorithm used in this work is an example of the application of this kind of filters to the task of edge detection.

Much slower than any of the previously cited, this type of algorithm reduces significantly the number of false-positives and false-negatives.

2.1.4.3.3 Colour gradient

All the above edge detection methods were applied to greyscale images. It was also decided to test a simple direct colour edge detection algorithm, based on the variations – gradient – of colour in every pixel and in every direction.

This method relied on the results achieved by applying the following expressions to every pixel in the image:

$$\nabla_{N-S} = \left|\mathrm{Red}_{(x-1,y)} \times \mathrm{Green}_{(x-1,y)} \times \mathrm{Blue}_{(x-1,y)} - \mathrm{Red}_{(x+1,y)} \times \mathrm{Green}_{(x+1,y)} \times \mathrm{Blue}_{(x+1,y)}\right| \quad (2\text{-}27)$$

$$\nabla_{E-W} = \left|\mathrm{Red}_{(x,y+1)} \times \mathrm{Green}_{(x,y+1)} \times \mathrm{Blue}_{(x,y+1)} - \mathrm{Red}_{(x,y-1)} \times \mathrm{Green}_{(x,y-1)} \times \mathrm{Blue}_{(x,y-1)}\right| \quad (2\text{-}28)$$

$$\nabla_{NW-SE} = \left|\mathrm{Red}_{(x-1,y-1)} \times \mathrm{Green}_{(x-1,y-1)} \times \mathrm{Blue}_{(x-1,y-1)} - \mathrm{Red}_{(x+1,y+1)} \times \mathrm{Green}_{(x+1,y+1)} \times \mathrm{Blue}_{(x+1,y+1)}\right| \quad (2\text{-}29)$$

$$\nabla_{NE-SW} = \left|\mathrm{Red}_{(x-1,y+1)} \times \mathrm{Green}_{(x-1,y+1)} \times \mathrm{Blue}_{(x-1,y+1)} - \mathrm{Red}_{(x+1,y-1)} \times \mathrm{Green}_{(x+1,y-1)} \times \mathrm{Blue}_{(x+1,y-1)}\right| \quad (2\text{-}30)$$

The maximum of these four values was then compared to a previously set threshold parameter. If its value was above the threshold, it became black; otherwise it remained white.

2.1.4.4 **Segmentation by Region detection**

For this purpose several types of algorithms were also applied; both on greyscale converted images and directly on the original colour pictures. Some of the most well known are described below.

2.1.4.4.1 Threshold iterative selection

The combined method used here and described by L. Xu et al. [70], can be seen as belonging to this group of algorithms.

In these methods, an initial value for the threshold is assumed. Its value is then refined by successive image processing steps [46].

In every step, the mean grey levels for all pixels below the threshold (T_b) and above it (T_a) are calculated and a new threshold is then calculated as the median value ($T_b + T_a$) / 2. The process stops when it is no longer possible to fine tune the average value between two passes.

If we assume that the picture histogram h is a one-dimensional array of fixed and small size and the initial threshold is T_0, then the *kth* estimated value for the threshold will be:

$$T_k = \frac{\sum_{i=0}^{T_k-1} i \cdot h[i]}{2\sum_{i=0}^{T_k-1} h[i]} + \frac{\sum_{j=T_{k-1}+1}^{N} j \cdot h[j]}{2\sum_{j=T_{k-1}+1}^{N} h[j]} \quad (2\text{-}31)$$

When $T_k = T_{k+1}$, T_k is the adequate threshold.

2.1.4.4.1.1 Entropy

The concept of Entropy [106,107] can be simply defined as a measure of information content. If we have n possible symbols x and if symbol i occurs with probability $p(x_i)$, the entropy - measured in bits/symbol - associated with the source of the symbols, X, is:

$$H(X) = -\sum_{i=1}^{n} p(x_i)\log(p(x_i)) \quad (2\text{-}32)$$

.

If we assume an image as a source of grey levels, the entropy associated with black pixels that result from the application of threshold t is:

$$H_b = -\sum_{i=0}^{t} p_i \log(p_i) \quad (2\text{-}33)$$

and with white pixels is:

$$H_w = -\sum_{i=t+1}^{255} p_i \log(p_i) \quad (2\text{-}34)$$

If the total entropy is given by:

$$H_T = -\sum_{i=0}^{255} p_i \log p_i \tag{2-35}$$

and

$$P_t = \sum_{i=0}^{t} p_i \tag{2-36}$$

is the probability that a given pixel will have a value less than or equal to *t*, according to Parker [46] the task now consists on maximizing:

$$f(t) = \frac{H_t}{H_T} \frac{\log P_t}{\log(\max(p_0, p_1, \ldots, p_t))} + \left(1 - \frac{H_t}{H_T}\right) \frac{\log(1 - P_t)}{\log(\max(p_{t+1}, p_{t+2}, \ldots, p_{255}))} \tag{2-37}$$

Zhang, Fritts and Goldman in their paper An Entropy-based Objective Evaluation Method for Image Segmentation [86] describe a method based on these concepts.

2.1.4.4.1.2 Minimum error thresholding

If we think of the image histogram as the measured probability density function of two regions' pixels – usually a normal distribution – then [46] it can be seen as an approximation to:

$$p(g) = \frac{1}{\sigma_1 \sqrt{2\pi}} e^{-\left((g-\mu_1)^2 / 2\sigma_1^2\right)} + \frac{1}{\sigma_2 \sqrt{2\pi}} e^{-\left((g-\mu_2)^2 / 2\sigma_2^2\right)} \tag{2-38}$$

where $\boldsymbol{\sigma}$ and $\boldsymbol{\mu}$ are the standard deviation and mean of both classes. This equation can be written as:

$$\frac{(g-\mu_1)^2}{\sigma_1^2} + \log \sigma_1 - 2 \log P_1 = \frac{(g-\mu_2)^2}{\sigma_2^2} + \log \sigma_2 - 2 \log P_2 \tag{2-39}$$

Since σ, μ and P are not known and difficult to estimate, Kittler and Illingworth [121] (1986) proposed that, given the following equations:

$$P_1(t)=\sum_{g=0}^{t}h(g) \qquad (2\text{-}40)$$

$$P_2(t)=\sum_{g=t+1}^{255}h(g) \qquad (2\text{-}41)$$

$$\mu_1(t)=\frac{\sum_{g=0}^{t}g\cdot h(g)}{P_1(t)} \qquad (2\text{-}42)$$

$$\mu_2(t)=\frac{\sum_{g=t+1}^{255}g\cdot h(g)}{P_2(t)} \qquad (2\text{-}43)$$

$$\sigma_1^2(t)=\frac{\sum_{g=0}^{t}h(g)(g-\mu_1(t))^2}{P_1(t)} \qquad (2\text{-}44)$$

$$\sigma_2^2(t)=\frac{\sum_{g=t+1}^{255}h(g)(g-\mu_2(t))^2}{P_2(t)} \qquad (2\text{-}45)$$

we should try to minimize the value:

$$J(t)=1+2(P_1(t)\log\sigma_1(t)+P_2(t)\log\sigma_2(t))-2(P_1(t)\log P_1(t)+P_2(t)\log P_2(t)) \qquad (2\text{-}46)$$

since the value of *t* that minimizes the above equation is the best threshold that can be used - normally called minimum error threshold [108,109,110,111].

Once again, the combined method used here [70] can be seen as an example of this type of segmentation algorithms.

2.1.4.4.2 Hue-Saturation thresholds

This method operates on the original colour image. It consists on defining maximum and minimum limits for Hue and Saturation. Every pixel on the image is then scanned and its components compared to those limits. If both the components have values between the limits, the pixel is set to white – normal skin. If any of the components values are outside the limits, it is set to the original image colour.

2.1.4.4.3 Hue-Saturation histogram thresholds

This is a method very similar to the last one. It uses the original colour image and by scanning every pixel on it, builds two histograms with the possible values for Hue and Saturation. Maxima and minima as well as the image mean values are extracted from the histograms. Both the values for Hue and Saturation kept on the histogram are only considered if their number of occurrences is, at least, a percentage of the maximum – input to the algorithm as a parameter.

2.2 Object recognition

During this part of the image recognition process, a class label [113,119] was assigned to the detected object [9], according to the previously identified features.

Several types of classifiers can be found in literature. All of them base their actions on an effective training phase. Since the expected results from a test set of images are already known – they must have been previously classified by an human expert - and since the most important goal, for such an application is the achievement of an accurate classification for every new analysed image, it was considered that the process of training the system should be supervised, keeping as primary objective, the reduction of a misclassification probability.

For this research the author focused on the kind of classifiers that support their actions on Bayes statistical theory, namely Naïve Bayes and Tree Augmented Naïve Bayes, and compared their results with those achieved by applying the same inputs to several other well known algorithms such as k-Nearest Neighbour, Fuzzy k-Nearest Neighbour, Support Vector Machines and Neural Networks – see descriptions for all these classifiers on section 8.3 - Appendix C.

All these algorithms are based on the concept of statistical learning. They all try to search for the most probable of the initially proposed hypothesis of an event, given a set of previously collected evidence. The first two of them are supported on Bayes rule of probabilities, as can be inferred from their names.

The Naïve Bayes model - see section 8.3.1 on Appendix C - assumes that all data representing the variables in the evidence set are fully independent. This allows for an a Maximum a Posteriori probability to be represented by the following expression:

$$\mathbf{P}(C|x_1, x_2, \dots, x_n) = \propto \mathbf{P}(C) \prod_i \mathbf{P}(x_i|C) \tag{2-47}$$

Where $x_1, x_2, \dots, x_n$ are the values of the various observed attribute variables and C is the unobservable class they belong to. This is a very straightforward method and one of the most

well behaved general purpose classification algorithms. It works well with large amounts of data and is not very affected by noise within the sample data.

Tree Augmented Naïve Bayes, a member of the more complex Bayesian Networks algorithms' class, assumes that the input variables are not necessarily independent from one another. To account for this, connections between variables are created. This usually leads to a more approximate fitting of the model to the real problem being analysed.

k-Nearest Neighbour algorithms described on section 8.3.2 on Appendix C, depend on a basic assumption: the characteristics of an input point are, quite likely, to be similar to its neighbour points.

To verify the level of similarity, every input feature vector is compared to the values of all the classes and it will be assigned to the class with more similarities. The performance of this type of algorithms is directly related to the number of neighbours processed; the bigger the number of neighbours, the better the classification. It is also related to the size of the training set. As the number of attributes in the input feature vectors and the number of samples increase, the computational effort grows significantly - $O(kN)^2$. Fuzzy *k*-Nearest Neighbour - see section 8.3.3 on Appendix C - algorithms are used to reduce the effect of noise in the input samples, albeit, at the expense of a more complex computational effort.

Artificial Neural Networks - see section 8.3.5 on Appendix C - try to emulate the way the human brain processes information. They are normally used for classification purposes. Although only applicable to linearly separable classes, usually, it will be possible to find a simple learning algorithm for solving this type of problems. For the type of classes within the scope of this work – two separable classes and several Boolean inputs – this is a kind of algorithms that perform in a very satisfactory way.

Support Vector machines belong to a class of algorithms known in the literature as kernel machines and whose description can be seen on section 8.3.4 on Appendix C. These are capable of using efficient training methods and also, accurately represent nonlinear classification functions.

Learning was achieved in a supervised way – see Appendix C. Images were previously classified by experts and the results achieved through the application of the author's automatic classification system were then compared to these previous classification values.

Unlike all the more or less similar systems analysed, this process has always been applied to unconstrained – non-calibrated – images. The Combined Method of edge detection described later was based on the work of L. Xu et al. [70] up to the stage of edge thinning. The next steps of the process were a combination of Helterbrand, J. D. [77] process with the application of rational Gaussian curve modelling [78] which results in more accurate edges.

Once the lesion's edges are accurately detected, several features can be extracted. These extracted features and mainly the way they were used during the classification process described below are, to the extent of my present knowledge, a totally novel approach.

3 Research process

This project was built upon all the work already done on the field of segmentation [45,96,97,98][72,115,116][73,99,100,101] [70] and pattern recognition [3], and it tries to achieve a classification rate of errors not greater than the average rate of a human expert diagnostic of skin tumours.

The first approach to the process consisted on the direct application of some elemental segmentation algorithms to previously converted greyscale images. The algorithms used were the following:

- Conversion to greyscale
    - The maximum value from Red, Green and Blue
    - The average value
    - NTSC luminance standard
    - Intel's image library formula [147]
- Segmentation by Edge Detection
    - Sobel [45,96,97,98]
    - Marr-Hildreth [72,115,116]
    - Canny [73,99,100,101]
    - Colour Gradient
- Segmentation by Region Detection
    - Hue-Saturation thresholds
    - Hue-Saturation histogram thresholds

The results, as can be seen below on sections 3.2.3 and 3.2.4 were not good enough. This led to more research and finally, to a whole new approach to the process, consisting on a rather complex combination of algorithms for edge detection, feature extraction and classification.

Since the segmentation results achieved with this set of techniques were rather accurate, the author decided to use it as the basis for the feature extraction process. The main steps, shown in Appendix H as a flowchart, were as follows:

- Gathering of images
- Feature Extraction
    - Conversion to greyscale
        - Initial conversion from RGB colourspace to XYZ colourspace
        - Final conversion from XYZ colourspace to CIELab colourspace
    - Segmentation by Edge Detection
        - Calculation of the standard deviation of the pixels of the image background
        - Smoothing of the image by the application of a Gaussian filter
        - Definition of a main threshold value, T
        - Definition of two hysteresis thresholds related to the above, T_1 and T_2
        - Application of these two latter thresholds to the smoothed image, resulting in bitmap B_1
        - Application of an edge detection Sobel filter to the originally smoothed image, giving another bitmap – B_2
        - Combination of B_1 and B_2 in order to get another bitmap – B_3 - with the points that are common to both
        - Creation of bitmap B_4 by segmenting the original smoothed greyscale image through the application of threshold T
        - Create another bitmap – B_5 – with an almost correct detected edge, by finding in B_2 the points orthogonal to the edge of B_4 that are closer to it
        - Thinning of the resulting edge by excluding some points' combinations that are not allowed
        - Closing of the edge by connecting every point on it to the nearest one by a line segment
        - Superimposition of the resulting edge on the original image
    - Calculation of the lesion's size
    - Calculation of the lesion's average radius

    - Determination of the level of jaggedness
        - Calculation of the probable centre of the lesion
        - Calculation of the standard deviation of the distances from the centre to every point on the lesion's edge
        - Calculation of the number of changes in the edge direction
    - Determination of the number of colours
        - Calculation of the standard deviation for the image pixels colours
        - Calculation of the standard deviations between the colours of adjacent pixels
    - Calculations of various ratios between previously determined values
- Training and testing of the system

3.1 Gathering of images

This was an ongoing task, from the very beginning until the end of all the text writing. Images both of skin tumours and of harmless skin spots were either collected from public Internet sites, or given to the author by dermatologists, from their own historical patients, with verified diagnoses, in order to create training and testing sets.

The feature extraction tasks are very dependent on the image characteristics. It would certainly be easier to guarantee that all the input images were captured and saved using a lossless format as the Portable Network Graphics (PNG) and thus, saving, restoring and re-saving an image would not degrade its quality. This work's intention though is to go beyond that ideal situation and so, when gathering the images, no restrictions have been imposed, either to image formats or resolutions – although most of them were lossy JPEG images.

Since the features considered for classification were based on ratios between values intrinsic to the image itself and not absolute quantities, these degrees of freedom are rather well dealt by the system. This is one of the major novelties introduced by this work and, the author believes, is one of the main reasons for the rather satisfactory results achieved by the overall process, regardless of the classification algorithm used.

3.2 Feature extraction

The first tasks consisted on extracting the features considered relevant from the images used to train and test the system.

3.2.1 "Normal" skin

The first set of extracted features was the imaging characteristics of "normal" skin. The images originally collected for the work of Cowell & Weston [71] and kindly made available for this research, were grouped by the following ethnic origins:

AC – African

CH – Chinese

IN – Indian

OA – Other Asian

WH - White

The work has been started by building a small Visual C++ program that allows the user to select a portion of an image, from a set of training images - and then stores the imaging data of a rectangle of marked pixels, as can be seen in figure 3-1, below.

From the data collected, values for Red, Green and Blue, as well as for Hue, Saturation and Brightness were kept – see Appendix A.

In order to reduce the influence of environmental conditions, the values for Red, Green and Blue were normalized and in fact, the values that were used were the ratios:

$$\frac{\text{Red}}{\text{Red} + \text{Green} + \text{Blue}} \quad (3\text{-}1) \qquad \frac{\text{Green}}{\text{Red} + \text{Green} + \text{Blue}} \quad (3\text{-}2) \qquad \frac{\text{Blue}}{\text{Red} + \text{Green} + \text{Blue}} \quad (3\text{-}3)$$

From all these values, averages, maxima and minima have been extracted for every reading – see Appendix B - and aggregated for every picture, as well as for the whole set of data – see tables below.

Since the Hue, Saturation and Brightness had such scattered values, it was decided not to consider them as an identification feature – see figures 3-2 and 3-3 on the following pages.

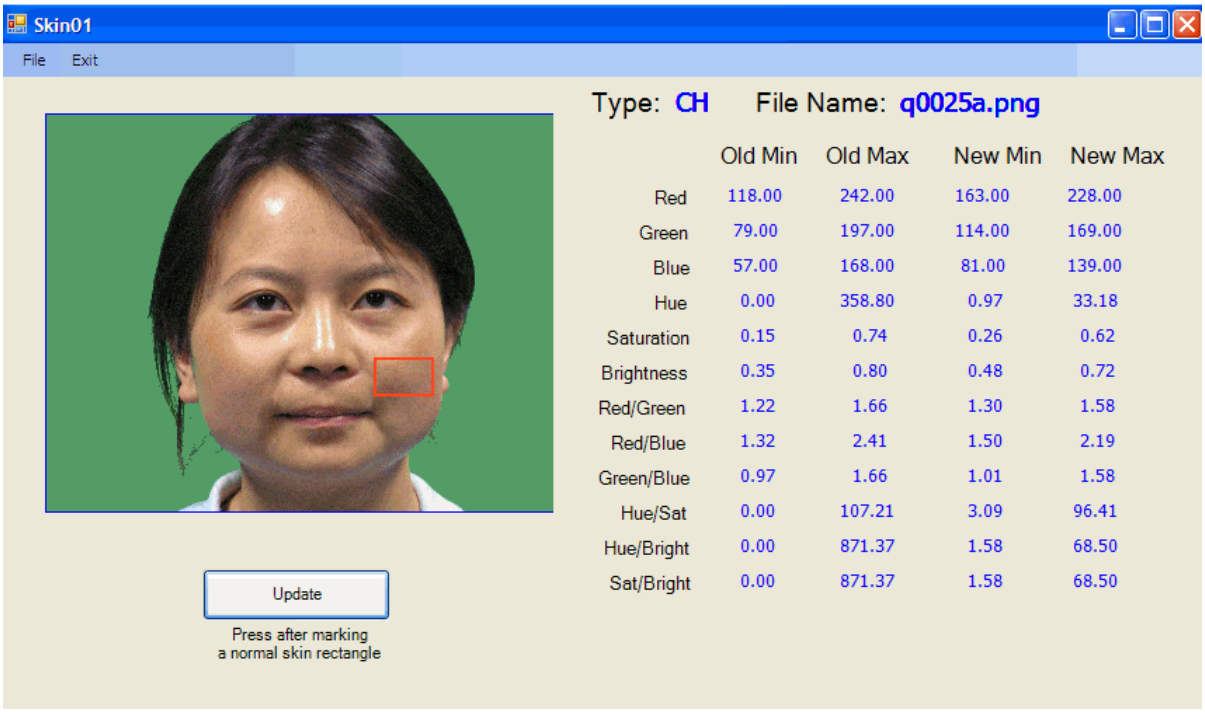


Figure 3-1 - Imaging data collection main window

Table 3-1 - Values for Hue, Saturation and Brightness of "White" skin

	Hue			Sat			Bri		
FileName	Min	Avg	Max	Min	Avg	Max	Min	Avg	Max
q0001a.png	0.56	12.14	246.71	0.36	0.62	0.83	0.55	0.67	0.81
q0004a.png	8.28	19.30	30.18	0.45	0.72	0.85	0.61	0.76	0.80
q0005a.png	2.21	12.70	138.92	0.28	0.43	0.55	0.51	0.64	0.72
q0006a.png	7.02	22.08	142.22	0.35	0.49	0.67	0.44	0.59	0.71
q0010a.png	4.05	17.31	144.49	0.27	0.44	0.75	0.49	0.61	0.79
q0011a.png	6.82	14.95	90.09	0.55	0.86	0.98	0.59	0.72	0.78
q0012a.png	3.76	12.95	91.18	0.53	0.71	0.84	0.55	0.70	0.78
q0013a.png	5.30	13.23	108.85	0.50	0.66	0.91	0.47	0.57	0.76
q0014a.png	3.28	16.16	194.35	0.24	0.39	0.67	0.46	0.61	0.76
q0016a.png	11.17	17.78	30.78	0.37	0.56	0.78	0.44	0.63	0.71
q0021a.png	0.00	10.49	359.30	0.34	0.62	0.92	0.49	0.71	0.83
q0031a.png	0.89	10.56	277.46	0.30	0.57	0.78	0.44	0.62	0.75
q0032a.png	3.51	17.21	249.72	0.27	0.39	0.52	0.45	0.60	0.65
q0034a.png	10.35	21.19	36.51	0.32	0.48	0.62	0.55	0.67	0.74
q0035a.png	3.82	18.39	26.95	0.34	0.56	0.87	0.49	0.65	0.78
q0036a.png	8.00	20.49	29.24	0.52	0.62	0.87	0.65	0.71	0.80
q0038a.png	0.60	13.79	248.69	0.35	0.59	0.74	0.56	0.73	0.80

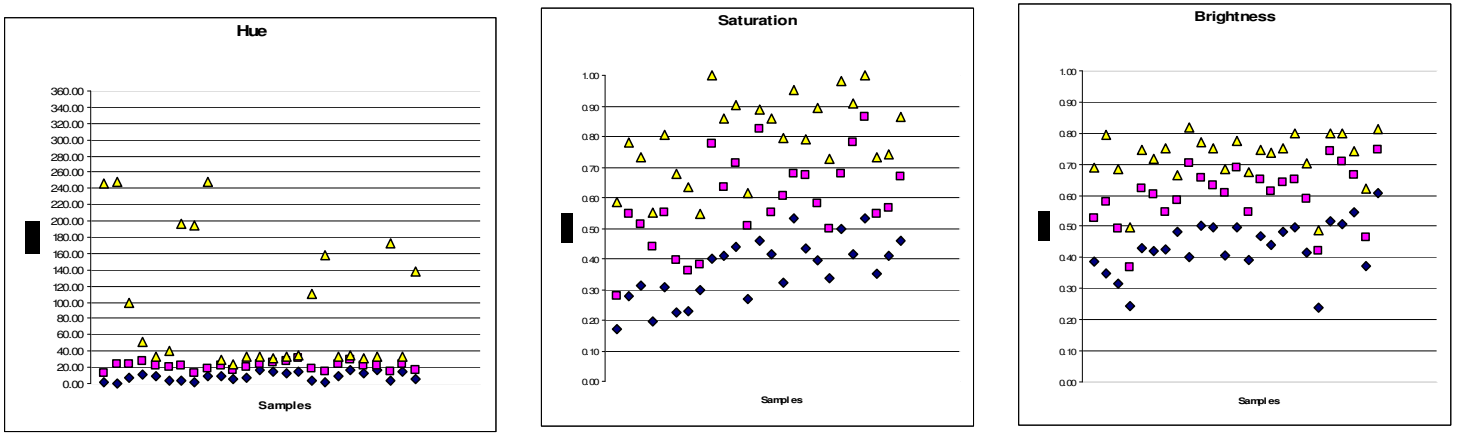


Figure 3-2 - "Non-White" skin Hue, Saturation and Brightness

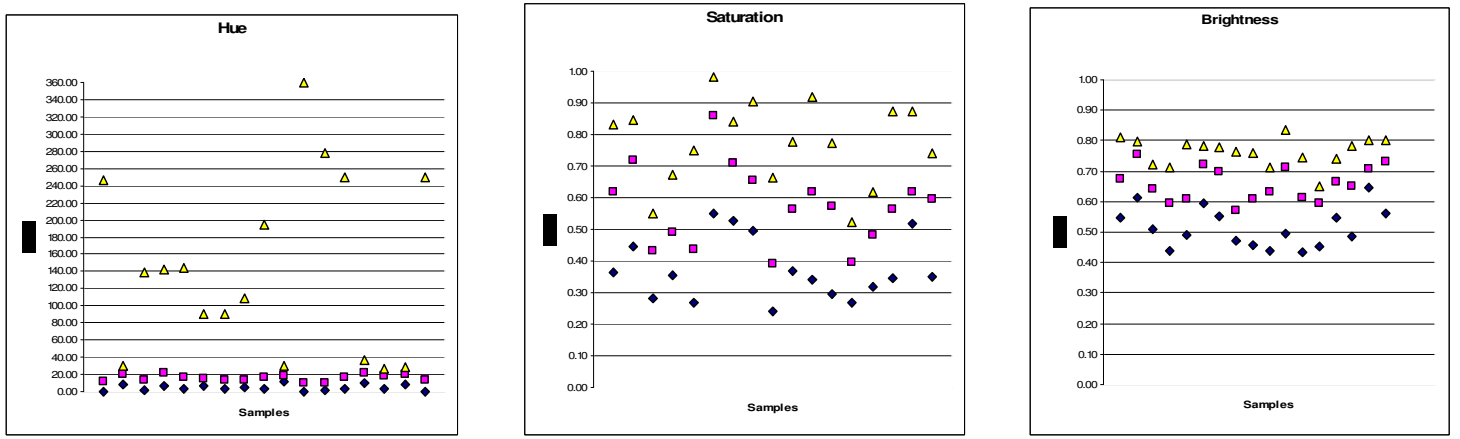


Figure 3-3 - "White" skin Hue, Saturation and Brightness

Table 3-2 - Values for Hue, Saturation and Brightness of "Non-White" skin

		Hue			Sat			Bri		
Type	**FileName**	**Min**	**Avg**	**Max**	**Min**	**Avg**	**Max**	**Min**	**Avg**	**Max**
AC	q0002a.png	1.75	13.32	246.59	0.17	0.28	0.58	0.39	0.53	0.69
AC	q0007a.png	0.59	24.71	247.31	0.28	0.55	0.78	0.35	0.58	0.79
AC	q0041a.png	7.77	23.17	100.02	0.31	0.51	0.73	0.32	0.49	0.68
AC	q0042a.png	10.71	27.71	51.51	0.20	0.44	0.55	0.24	0.37	0.50
CH	q0023a.png	9.98	21.24	33.57	0.31	0.55	0.81	0.43	0.62	0.75
CH	q0024a.png	3.78	19.49	39.62	0.22	0.40	0.68	0.42	0.60	0.72
CH	q0025a.png	4.37	21.74	197.22	0.23	0.36	0.64	0.43	0.55	0.75
CH	q0026a.png	1.21	12.50	193.96	0.30	0.38	0.55	0.49	0.58	0.67
IN	q0008a.png	8.73	18.06	248.51	0.40	0.77	1.00	0.40	0.70	0.82
IN	q0015a.png	9.18	21.68	29.84	0.41	0.64	0.86	0.50	0.65	0.77
IN	q0018a.png	6.09	16.31	24.74	0.44	0.71	0.90	0.50	0.63	0.75
IN	q0019a.png	7.10	20.00	32.23	0.27	0.51	0.62	0.41	0.61	0.68
IN	q0029a.png	17.05	24.25	33.86	0.46	0.82	0.89	0.50	0.69	0.77
IN	q0030a.png	14.09	24.93	32.11	0.42	0.55	0.86	0.39	0.54	0.67
IN	q0039a.png	13.46	27.28	33.50	0.32	0.60	0.80	0.47	0.65	0.75
OA	q0003a.png	14.49	31.11	35.72	0.53	0.68	0.95	0.44	0.61	0.74
OA	q0009a.png	3.97	17.60	109.58	0.43	0.67	0.79	0.49	0.64	0.75
OA	q0017a.png	2.30	14.15	157.75	0.39	0.58	0.89	0.50	0.65	0.80
OA	q0020a.png	8.40	23.36	33.03	0.34	0.50	0.73	0.42	0.59	0.70
OA	q0022a.png	15.98	29.36	35.74	0.50	0.68	0.98	0.24	0.42	0.49
OA	q0027a.png	13.71	22.58	31.32	0.42	0.78	0.91	0.52	0.74	0.80
OA	q0028a.png	16.80	22.90	32.28	0.53	0.86	1.00	0.51	0.71	0.80
OA	q0033a.png	4.10	15.24	171.82	0.35	0.55	0.73	0.55	0.67	0.74
OA	q0037a.png	15.13	24.12	33.71	0.41	0.57	0.74	0.37	0.46	0.62
OA	q0040a.png	5.69	16.44	137.66	0.46	0.67	0.86	0.61	0.75	0.81

After these results, work has been focused on RGB parameters.

At a first glance, these values looked more promising, In fact, after normalized, the values were quite "concentrated" – see Tables 3-3 and 3-4 and Figures 3-4 and 3-5 below. Nevertheless, after a more thorough analysis, it was verified that the relations for Red, Green and Blue were very similar for all the types of skin.

Table 3-3 - Values for Red, Green and Blue of "Non-White" skin

FileName	Avg of Rmin	Avg of Ravg	Avg of Rmax	Avg of Gmin	Avg of Gavg	Avg of Gmax	Avg of Bmin	Avg of Bavg	Avg of Bmax
q0002a.png	0.45	0.43	0.41	0.33	0.31	0.31	0.23	0.27	0.27
q0003a.png	0.54	0.47	0.45	0.29	0.33	0.34	0.16	0.20	0.22
q0007a.png	0.51	0.44	0.40	0.30	0.32	0.32	0.20	0.24	0.28
q0008a.png	0.47	0.44	0.42	0.30	0.31	0.32	0.23	0.25	0.26
q0009a.png	0.50	0.45	0.43	0.29	0.30	0.31	0.21	0.24	0.27
q0015a.png	0.48	0.45	0.43	0.31	0.31	0.31	0.22	0.24	0.26
q0017a.png	0.50	0.46	0.42	0.27	0.30	0.30	0.23	0.25	0.27
q0018a.png	0.49	0.47	0.45	0.30	0.30	0.30	0.22	0.23	0.25
q0019a.png	0.47	0.45	0.43	0.32	0.31	0.31	0.21	0.24	0.26
q0020a.png	0.50	0.45	0.44	0.31	0.32	0.31	0.19	0.23	0.25
q0022a.png	0.69	0.54	0.52	0.30	0.33	0.32	0.01	0.12	0.16
q0023a.png	0.49	0.46	0.43	0.31	0.31	0.32	0.20	0.23	0.25
q0024a.png	0.44	0.43	0.40	0.32	0.31	0.33	0.24	0.26	0.27
q0025a.png	0.45	0.43	0.40	0.33	0.32	0.32	0.23	0.25	0.28
q0026a.png	0.48	0.45	0.43	0.30	0.30	0.30	0.22	0.26	0.27
q0027a.png	0.48	0.43	0.41	0.32	0.32	0.32	0.20	0.25	0.27
q0028a.png	0.53	0.47	0.42	0.31	0.31	0.33	0.16	0.22	0.25
q0029a.png	0.50	0.46	0.42	0.31	0.32	0.33	0.19	0.22	0.25
q0030a.png	0.53	0.49	0.45	0.32	0.32	0.33	0.15	0.19	0.22
q0033a.png	0.41	0.44	0.43	0.33	0.30	0.30	0.26	0.25	0.27
q0037a.png	0.56	0.52	0.47	0.31	0.31	0.32	0.12	0.17	0.21
q0039a.png	0.48	0.43	0.42	0.32	0.33	0.32	0.20	0.24	0.26
q0040a.png	0.45	0.42	0.41	0.31	0.31	0.31	0.24	0.27	0.28
q0041a.png	0.55	0.49	0.43	0.32	0.31	0.32	0.14	0.20	0.25
q0042a.png	0.49	0.48	0.43	0.36	0.33	0.33	0.16	0.19	0.24

Table 3-4 - Values for Red, Green and Blue of "White" skin

FileName	Avg of Rmin	Avg of Ravg	Avg of Rmax	Avg of Gmin	Avg of Gavg	Avg of Gmax	Avg of Bmin	Avg of Bavg	Avg of Bmax
q0001a.png	0.46	0.44	0.41	0.30	0.30	0.31	0.24	0.26	0.29
q0004a.png	0.44	0.42	0.41	0.32	0.32	0.31	0.24	0.27	0.28
q0005a.png	0.45	0.43	0.41	0.32	0.30	0.31	0.24	0.27	0.28
q0006a.png	0.51	0.45	0.41	0.29	0.31	0.32	0.20	0.23	0.27
q0010a.png	0.45	0.43	0.40	0.32	0.31	0.32	0.23	0.26	0.28
q0011a.png	0.50	0.47	0.44	0.28	0.29	0.30	0.22	0.24	0.26
q0012a.png	0.50	0.45	0.43	0.28	0.29	0.30	0.22	0.25	0.27
q0013a.png	0.55	0.51	0.43	0.27	0.27	0.31	0.18	0.21	0.26
q0014a.png	0.45	0.41	0.40	0.31	0.31	0.31	0.25	0.27	0.28
q0016a.png	0.51	0.46	0.44	0.31	0.30	0.31	0.19	0.24	0.25
q0021a.png	0.49	0.44	0.41	0.28	0.30	0.30	0.22	0.27	0.29
q0031a.png	0.51	0.46	0.42	0.27	0.29	0.31	0.22	0.25	0.27
q0032a.png	0.47	0.43	0.43	0.29	0.31	0.31	0.25	0.26	0.27
q0034a.png	0.44	0.42	0.41	0.32	0.32	0.31	0.24	0.26	0.28
q0035a.png	0.47	0.44	0.42	0.31	0.31	0.30	0.22	0.25	0.27
q0036a.png	0.45	0.43	0.41	0.31	0.32	0.32	0.24	0.25	0.27
q0038a.png	0.45	0.42	0.40	0.31	0.30	0.31	0.25	0.27	0.29

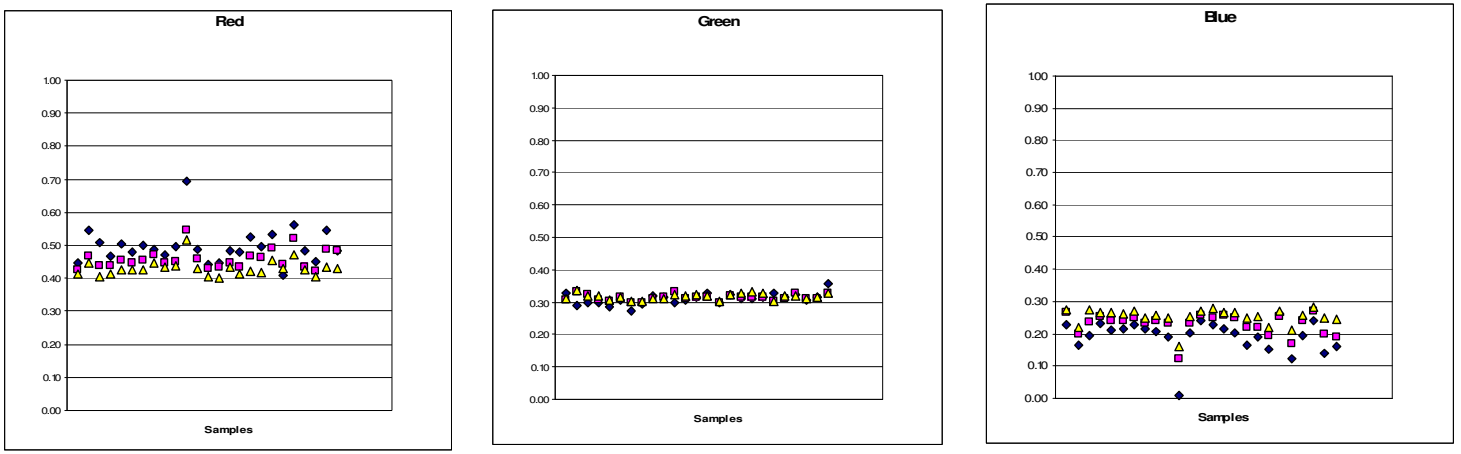


Figure 3-4 - "Non- White" skin Red, Green and Blue

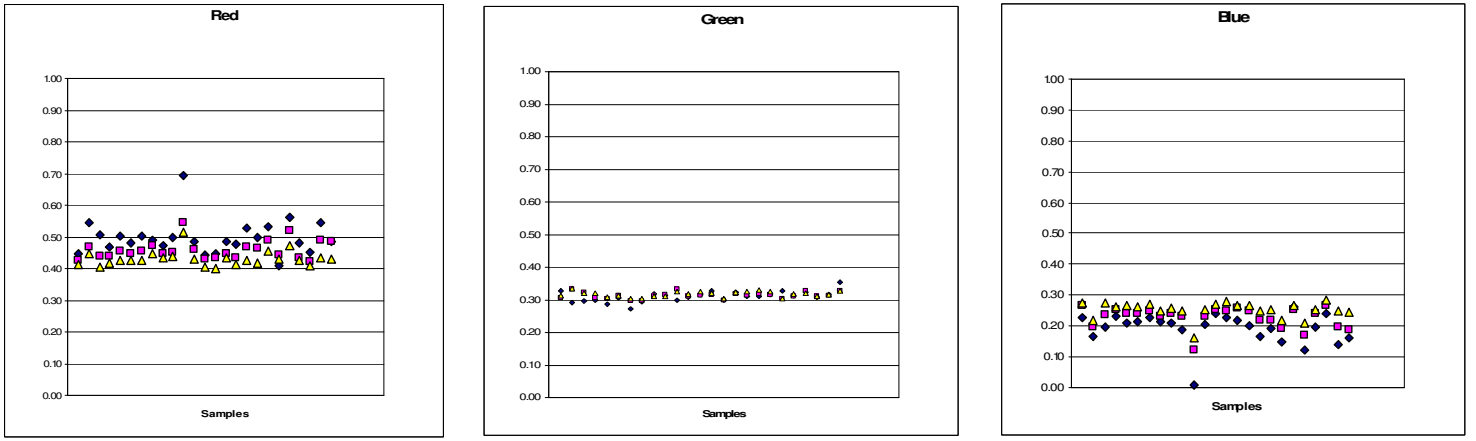


Figure 3-5 - "White" skin Red, Green and Blue

After this and with the help of medical professionals, all the features described in section 1.2 were evaluated as to their importance for the classification task.

For achieving this goal, a set of skin spot's images was processed and data has been extracted using one or several of the algorithms described below.

It has then, been possible to use an application that, after receiving the same information from a new set of pictures, classified the collected data as belonging to the base class or not. The results were then evaluated – in an iterative process - in order to fine tune the initial boundaries.

The whole process of training and testing was repeated as long as the test results became more satisfactory than the predecessors or, the system performance improvements stopped at a point that did not allow any further significant improvement.

Every value was stored on a file for every specific image, so that evolution can be followed.

The detected edges were superimposed on the original image and the resultant bitmap was saved as a png [122,123] format image.

Assuming – as seems obvious – that there will, almost always, be some kind of asymmetry on the tested skin spots, two values were calculated; one relative to the biggest dimension and another one orthogonal to it. To be able to get those values from a skin spot image like the one that can be seen in figure 3-6, the edge of the spot had to be identified. This task was accomplished through the use of one of the most used and already described algorithms for edge and region detection [4,45,46].

The detection of edges, allows for the definition of every other relevant feature.

Figure 3-6 - Melanoma

By running a program similar to the one that was used to extract normal skin imaging characteristics, the author has been able to extract values for HSB or/and normalized RGB values, from which the standard deviation for the colour of the pixels inside the mole region can be calculated.

Having defined the border of the mole, its size is determined by a small program that counts the number of pixels inside its borders. The difference in sizes defines the percentage of growth between two points in time.

Since adequate software could not be found, the first task was the development of programs that could extract the relevant features from the original images of the gathered skin moles, namely, segmentation/edge detection. From there it has been possible to define the mole's size, asymmetry and colour variations.

3.2.2 Initial conversion to greyscale

The first program developed addressed the colour to greyscale images conversion [119], with the purpose of applying some of the most known edge detection algorithms on them. The algorithms used were those described in section 2.1.4.1.

The images below are examples of the results obtained by applying these algorithms to the image represented in figure 3-6. Since the intention was to achieve high level of variations between intensities of grey, on the edges of the lesions, it was decided to use the images obtained by application of the first algorithm represented - maximum of the values of Red, Green and Blue image components.

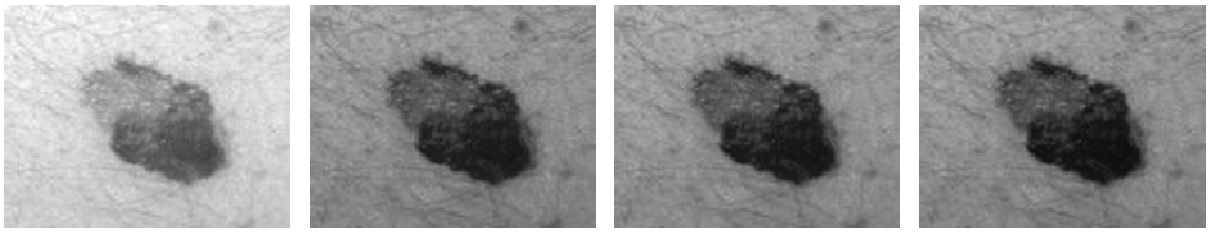


Figure 3-7 - Maximum, Average, NTSC and Intel greyscale images

3.2.3 Edge detection

To accomplish this task the following algorithms were implemented:

- Sobel
- Marr-Hildreth
- Canny
- Colour gradient analysis

Since it was not possible to find a software product that would do exactly what was needed to fulfill the proposed objectives, programs were developed that extracted all the features, a priori, considered relevant to the automatic classification of the moles. The main panel for the first program can be seen in figure 3-8.

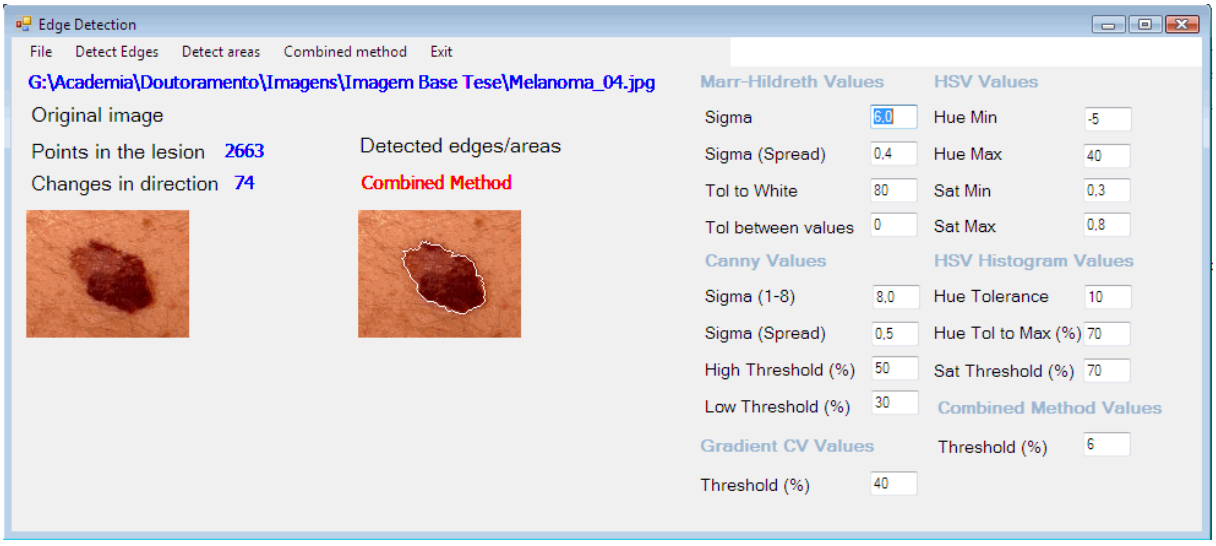


Figure 3-8 - Main edge detection program's panel

3.2.3.1 Sobel edge detection

The Sobel edge detection algorithm [46] is one of the most well known – and simple - template based edge detection algorithm. It operates by calculating intensity gradients for every pixel in the image and applying to every one of them, two templates - *convolution masks* - having the following values:

$$S_x = \begin{matrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{matrix}$$

and

$$S_y = \begin{matrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{matrix}$$

These masks can be seen as an approximation to the gradient at the pixel corresponding to the point at their centre. Assuming all this and for a pixel at location (i, j), S_x and S_y can be computed by:

$$\begin{aligned} S_x = {} & I[i-1][j+1] + 2I[i][j+1] + I[i+1][j+1] \\ & - (I[i-1][j-1] + 2I[i][j-1] + I[i+1][j-1]) \end{aligned} \tag{3-4}$$

$$\begin{aligned} S_y = {} & I[i+1][j+1] + 2I[i+1][j] + I[i+1][j-1] \\ & - (I[i-1][j+1] + 2I[i-1][j] + I[i-1][j-1]) \end{aligned} \tag{3-5}$$

After the S_x and S_y components are calculated for every pixel on the image, the result must be compared to the predefined threshold. Pixels with gradient intensities above that value will be written as black – edge pixel – and every other will be changed to white.

In this work and for this type of filter, the threshold was dynamically determined as the value in the middle of the gradient levels with relevant occupation by image pixels. This has been achieved by building an image intensities histogram. T2, the upper threshold was determined as the highest intensity measured in anyone of the image pixels. T1, the lowest threshold value was calculated by applying the input percentage to the value of T2.

For the image in figure 3-6 above, the results of the application of the Sobel filter, considering only intensities between the one that was most often detected and levels of grey with at least 0.5%, 1%, 2.5%, 5%, 10 and 20% of that number of pixels are represented on the next images.

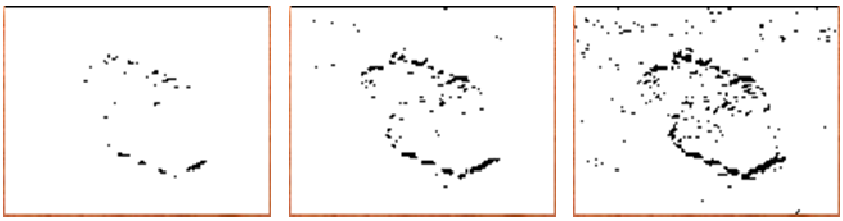


Figure 3-9 - Sobel filter with 0.5%, 1% and 2.5% low threshold

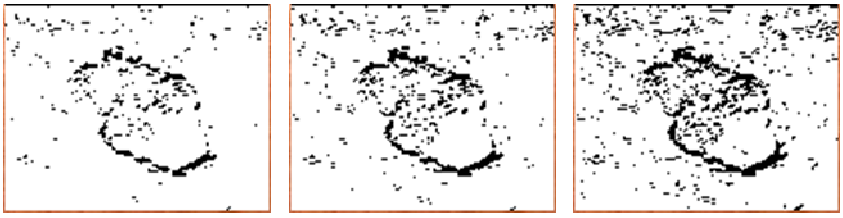


Figure 3-10 - Sobel filter with 5%, 10% and 20% low threshold

From the above images, the lack of precision of this filter becomes evident. If the low threshold is too low, the filter delivers many false negatives. As the threshold grows in percentage of the maximum number of occurrences of one intensity level, the number of false positives grows along with it, resulting in a quite low precision detected edge.

3.2.3.2 Marr-Hildreth edge detection

This method, presented by the authors in 1979 [72] is based on assumptions such as:

- In *"natural images, changes can and do occur over a wide range of scales"* and so, they must also be dealt with in different ways. This led to the need for using different operators and combining their results;

- This requirement brings us to a situation where it is essential to define "*local averages of the image at different resolutions*" and after that, detecting "*the changes in intensity that occur at*

each one". To achieve this, and consequently, being able to computationally detect intensity changes – edges - it is imperative that their range of scales be reduced. This corresponds to the implementation of a smoothing filter so that its resulting image spectrum should be formed by band-limited frequencies, that is, the variance between intensities within the image should be as small as possible;

- According to Marr and Hildreth, the reasons for intensity changes in the image are

    - Illumination changes;
    - Changes in the orientation or distance of the viewer to the visible surfaces;
    - Change in surface reflectance

  The contribution to each point in the filtered image should then be the result of a smooth average of its neighbour points.
  The filter that best satisfies all these conditions is the Gaussian, given for the image two dimensions by the following expression where $r = \sqrt{x^2 + y^2}$:

$$G(x,y) = G(r) = \frac{1}{2\pi\sigma^2} e^{(-r^2/2\sigma^2)} \tag{3-6}$$

- Also, when an edge is found – which results in an abrupt change in intensity - there is an extreme high value for its first derivative, corresponding to a zero crossing situation in the second derivative;

  "*The intensity variation near and parallel to the line of zero-crossings should locally be linear*" [72] and if it can be calculated without special concern about direction, the needed computation effort will be much reduced. "*The only orientation-independent second order differential operator is the Laplacian*". Given by:

$$\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \tag{3-7}$$

  which, on a discrete environment – as an image is – and applied to the Gaussian becomes:

$$\nabla^2 G_\sigma = \left(\frac{r^2 - 2\sigma^2}{\sigma^4}\right) e^{\left(\frac{-r}{2\sigma^2}\right)} \quad (3\text{-}8)$$

3.2.3.2.1 Marr-Hildreth Algorithm

The algorithm will then be described as:

1. Execute the convolution of the original image with the Gaussian filter function;

$$I * G(x,y) = \sum_i \sum_j I(i,j) G(x-i, y-j) \quad (3\text{-}9)$$

2. Calculate the Laplacian of the Gaussian [124,125] filtered image;

3. Extract the pixels where zero-crossings are found.

In this work and as way to address different frequency scales, two Gaussians were calculated and each one of them was applied to the original image. The edge selected pixels were those that had zero-crossing values in both the approaches.

Values of the standard deviation - σ - between 4 and 8 have been tested. Since two values of σ were needed for every computation, the final values were calculated applying a spread to the original number. As an example and for an original σ of **4** and a spread of **0.2**, the values used for computation purposes are $\boldsymbol{\sigma_1 = 3.8}$ and $\boldsymbol{\sigma_2 = 4.2}$.

3.2.3.2.2 Results

As can be seen by the next images, the results were, nevertheless, not good enough for the objectives of this work.

Figure 3-11 - Marr-Hildreth edges detected with sigma=4 and spreads between 0.1 and 0.5

Figure 3-12 - Marr-Hildreth edges detected with sigma=6 and spreads between 0.1 and 0.5

Figure 3-13 - Marr-Hildreth edges detected with sigma=8 and spreads between 0.1 and 0.5

Figure 3-14 - Marr-Hildreth edges detected with spread=0.1 and sigmas between 4 and 8

3.2.3.3 Canny edge detector

In the year of 1986, John Canny [73] argued that a good edge detection algorithm should respect three performance criteria:

1. **Good Detection** [73]

   Real edges should, with high probability, be detected and non-real edge points should have a low probability of being wrongly detected. These probabilities are directly proportional to the signal-to-noise ratio and so, it can be achieved by maximizing this relation.

Assuming that:

- The filter's impulse response to and edge $G(x)$ is the function $f(x)$
- The edge is centred at a point where $x = 0$
- The filter has a finite response bounded by $[-W, W]$
- n_0^2 is the mean-squared noise amplitude per unit length

then, the response of the filter to the edge at its centre is given by the convolution integral:

$$H_G = \int_{-W}^{+W} G(-x) f(x) dx \tag{3-10}$$

The root-mean-squared filter's response to the noise $n(x)$ can be described by:

$$H_n = n_0 \sqrt{\left[\int_{-W}^{+W} f^2(x) dx\right]} \tag{3-11}$$

and so, this first criterion can be expressed as:

$$SNR = \frac{H_G}{H_n} = \frac{\int_{-W}^{+W} G(-x) f(x) dx}{n_0 \sqrt{\left[\int_{-W}^{+W} f^2(x) dx\right]}} \tag{3-12}$$

2. **Good localization** [73]

   All the points referenced as being edges should be as close to the centre of the real edge as possible. To be effective, the measure for this criterion must increase as localization itself, increases. This measure was achieved through the use of the reciprocal of the root-mean-squared distance between the marked edge point and the real edge point. Since edges will result in local maxima for the filter output, its first derivative at those points will be zero. Because edges are centred at $x = 0$ and assuming that there is no noise to be considered at that location, x = 0 should correspond to a local maximum.

If $H_n(x)$ is the response of the filter to noise only, being $H_G(x)$ its response to the edge, and assuming there is a local maximum for $x = x_0$, we shall have:

$$H'_n(x_0) + H'_G(x_0) = 0 \tag{3-13}$$

If we consider the point where $x = 0$, and the Taylor series [127] expansion at this point – Maclaurin series [126] – in generic terms, given by:

$$f(x) = f(0) + \frac{f'(0)}{1!}x + \cdots \tag{3-14}$$

Since x_0 is supposed to be a very small displacement relative to the real edge point, the terms of higher order are negligible - applying the series expansion to $H'_G(x_0)$ we get:

$$H'_G(x_0) = H'_G(0) + H''_G(0)x_0 + \cdots \tag{3-15}$$

We assumed that the filter's response, at the origin and in the absence of noise, has a local maximum. Being so, $H'_G(0)$ can be ignored.

From the equations above, results that:

$$H'_n(x_0) + H''_G(0)x_0 = 0 \tag{3-16}$$

and

$$H''_G(0)x_0 \approx -H'_n(x_0) \tag{3-17}$$

Because the noise is Gaussian and has a variance that is given by the expectation of the mean-squared value of $H'_n(x_0)$:

$$E[H'_n(x_0)^2] = n_0^2 \int_{-W}^{+W} f'^2(x)dx \tag{3-18}$$

we get:

$$E[x_0^2] \approx \frac{n_0^2 \int_{-W}^{+W} f'^2(x)dx}{\left[\int_{-W}^{W} G'(-x)f'(x)dx\right]^2} = \partial x_0^2 \tag{3-19}$$

The Localization criterion is then the reciprocal of this value and is given by:

$$\text{Localization} = \frac{\left|\int_{-W}^{+W} G'(-x)f'(x)dx\right|}{n_0\sqrt{\int_{-W}^{+W} f'^2(x)dx}} \tag{3-20}$$

The objective of the whole process is to find the values that maximize both these two criteria. To achieve this we combine them both and try to maximize their product:

$$\frac{\int_{-W}^{+W} G(-x)f(x)dx}{n_0\sqrt{\left[\int_{-W}^{+W} f^2(x)dx\right]}} \frac{\left|\int_{-W}^{+W} G'(-x)f'(x)dx\right|}{n_0\sqrt{\int_{-W}^{+W} f'^2(x)dx}} \tag{3-21}$$

3. **Only one response to a single edge** [73]

This point can be seen as a special case of the first rule. If an edge is detected more than once, then one of the resulting points must be a false edge point. This criterion is necessary since the two first only addressed values at the centre of the edge, disregarding every pixel nearby.

If we consider Cauchy–Schwarz inequality, also known as the Schwarz inequality, the Cauchy inequality, or the Cauchy–Schwarz–Bunyakovsky inequality:

$$\left[\int_a^b \boldsymbol{f(x)g(x)dx}\right]^2 \leq \int_a^b \boldsymbol{f^2(x)dx} \int_a^b \boldsymbol{g^2(x)dx} \tag{3-22}$$

the Signal to Noise ratio (SNR) maximum is given by:

$$\boldsymbol{SNR_{max} = n_0^{-1}\sqrt{\int_{-W}^{+W} G^2(x)dx}} \tag{3-23}$$

and the Localization maximum is:

$$\textbf{Localization}_{max} = \boldsymbol{n_0^{-1}} \sqrt{\int_{-W}^{+W} \boldsymbol{G'^2(x)dx}} \tag{3-24}$$

These values are both maximized when $\boldsymbol{f(x) = G(-x)}$ within the interval $\boldsymbol{[-W, +W]}$.

Since the image will, inevitably, have a component of noise, and because there will be some interaction between neighbour points, it is probable that such a filter will get several positive responses around the real edge. The problem becomes being able to separate both responses in such a way that the probability that the filter accepts two of them is small enough to allow for the detection of only one edge point. This can be achieved if the distance between noise peaks is known. In fact, the difference between two high values of noise should, at least, be slightly bigger than the width needed to accommodate the response of the operator. Given that the average distance between zero-crossings of the response of a function g to Gaussian noise is [74]:

$$x_{avg} = \pi \sqrt{\frac{-R(0)}{R''(0)}} \tag{3-25}$$

where $R(\tau)$ is the autocorrelation function of g. Because $R(0)$ is given by:

$$R(0) = \int_{-\infty}^{+\infty} g^2\,(x)dx \tag{3-26}$$

and

$$R''(0) = -\int_{-\infty}^{+\infty} g'^2\,(x)dx \tag{3-27}$$

the mean distance between zero-crossings of f' can then be written as:

$$\boldsymbol{x_{zc}}(f) = \pi \sqrt{\frac{\int_{-\infty}^{+\infty} f'^2(x)dx}{\int_{-\infty}^{+\infty} f''^2(x)dx}} \tag{3-28}$$

The distance between adjacent maxima in the noise response, $x_{\max}$, is given:

$$x_{max}(f) = 2x_{zc}(f) \tag{3-29}$$

An efficient and computationally less demanding approximation to the problem stated by Canny is, in fact, the first derivative of the Gaussian function:

$$G(x) = e^{-\frac{x^2}{2\sigma^2}} \tag{3-30}$$

whose first derivative is:

$$G'(x) = \left(-\frac{x}{\sigma^2}\right) e^{-\left(\frac{x^2}{2\sigma^2}\right)} \tag{3-31}$$

For two dimensions the Gaussian function becomes:

$$G(x,y) = \sigma^2 e^{-\left(\frac{x^2+y^2}{2\sigma^2}\right)} \tag{3-32}$$

and

$$G_n(x,y) = \frac{\partial G}{\partial n} = n \cdot \nabla G \tag{3-33}$$

The direction of the gradient vector should be normal to the edge. Not knowing this direction *a priori*, nevertheless, a good estimation can be made by the smoothed gradient direction:

$$n = \frac{\nabla(G \times I)}{|\nabla(G \times I)|} \tag{3-34}$$

An edge point is then defined as a local maximum of G_n in the direction of n and applied to the image I and so:

$$\frac{\partial}{\partial n} G_n \times I = 0 \tag{3-35}$$

Substituting G_n by the gradient operator described above, we get:

$$\frac{\partial^2}{\partial n^2} G_n \times I = 0 \tag{3-36}$$

This convolution with a two dimensional image can be divided in two separate convolutions along one of the axis. An edge point defined by this equation will have its magnitude given by:

$$|G_n \times I| = |\nabla(G \times I)| \tag{3-37}$$

The Canny algorithm for edge detection should consequently be composed by the following actions:

- Read the image to process, I;
- Create a mask G - a one dimension Gaussian – to convolve with I, according to a given standard deviation – σ;
- With the same standard deviation value, create two other masks, G_x and G_y, for the first derivative of the Gaussian along both directions – x and y;
- Convolve the original image, I, with G along both directions in order to obtain I_x and I_y;
- Execute the convolution of I_x with G_x and of I_y with G_y, to get I'_x and I'_y;
- Combine both results the x and y components. Its magnitude at each pixel (x,y) shall be calculated as follows:

$$M(x,y) = \sqrt{I'_x(x,y)^2 + I'_y(x,y)^2} \quad (3\text{-}38)$$

- To the resulting points should then be applied a previously defined magnitude threshold. Those pixels with magnitude greater than the threshold value will be black and all others will remain white.

The results achieved by applying this filter to the image already used to demonstrate other filters outputs are shown in figure 3.15:

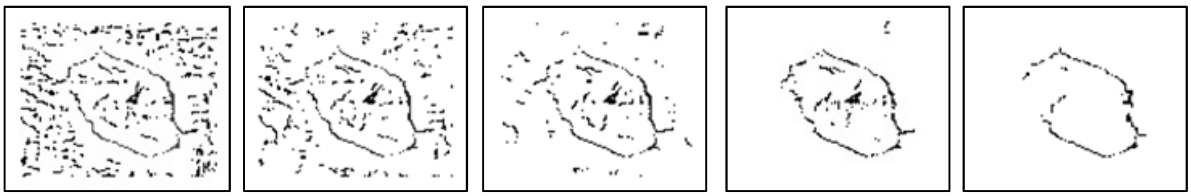


Figure 3-15 - Canny edges detected with sigma = 4, 5, 6 7 and 8

3.2.3.4 Colour gradient

Melanomas and in general, malignant skin tumours usually do not have a regular colour. The variations in colour, once the lesion area is identified, can be achieved through the calculation of colour values' standard deviation combined by the total colour difference between adjacent points. Once again, the problem resumes to the definition of a threshold above which there is a high probability of the tumour being a malignant one.

As said before, since the results obtained by the direct application of the above - some of the most known -algorithms for edge detection was not satisfactory – at least for the objectives pursued on this work – it was decided to try a simpler approach. Instead of converting the image to grey level, all the actions were applied on the original colour image.

The task consisted on, for every pixel on the image, calculating the gradient magnitude of the total colour values – product of the Red, Green and Blue components - along the four possible directions, N-S, E-W, NW-SE and NE-SW.

The results, shown in fig. 3-16, were interesting but still not good enough for the goal of this work.

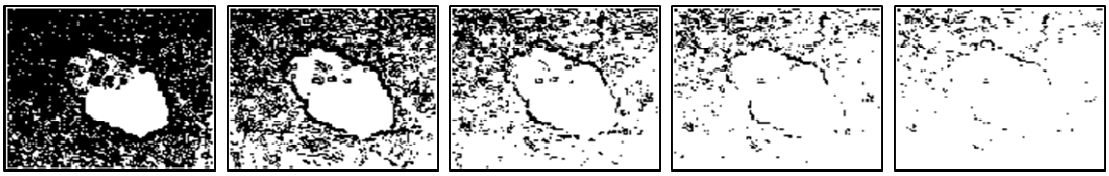


Figure 3-16 - Colour gradient edge detection with thresholds = 10, 20, 30, 40 and 50

3.2.4 Region detection

After these edge detection tests – and because all the results were not good enough – the segmentation approach in the original image over the Hue-Saturation-Value (Brightness) colour space was tried. Two different methods were tested:

- Hue-Saturation thresholds (see section 2.1.4.4.2).
- Hue-Saturation histogram thresholds (see section 2.1.4.4.3).

3.2.4.1 Hue-Saturation thresholds

The images resulting from the application of this method to the original picture are shown in figure 3-17.

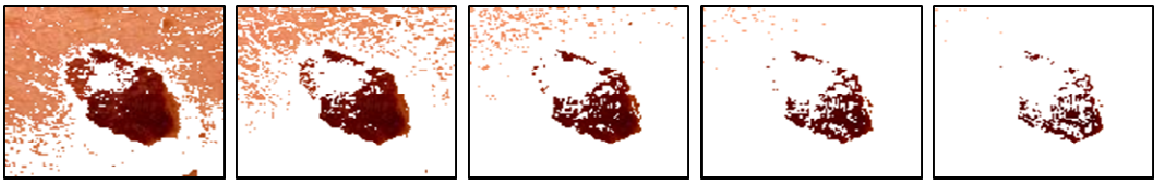


Figure 3-17 - Hue-Saturation areas with Hue_min = -5, Hue_Max = 70, Sat_min = 0.1 and Sat_Max = 0.6, 0.7, 0.8, 0.9 and 1.0

3.2.4.2 Hue-Saturation histogram thresholds

This is a method gave results – shown in figure 3-18 – that, although better than those achieved with the former segmentation filter, are still very rough for the objectives of this work.

Figure 3-18 - Hue-Sat Histogram areas with Hue threshold = 70% of the maximum Hue histogram value and Sat = 40%, 50%, 60%, 70% and 80% of the maximum Saturation histogram value

3.2.5 Combined method

During the research process, a paper by L. Xu et al. [70] on this exact subject called for the attention of the author. After thorough analysis it was then used as a framework for this combined method. Although not exactly equal to the method implemented here, it allowed for the building of a skeleton on which to support every other piece of the whole process.

Since the final edge is to be written down as a white line, to ensure that points external to it would not be recognized as belonging to the edge, the first operation consisted on checking every pixel of the image and if all the image RGB components had a value of 255, they were all changed to 254.

3.2.5.1 Conversion to CIELab colourspace

The second step on the method is the conversion of the colour image to the CIELAB colour space [75,129]. This colour space is a result of the work by the International Commission on Illumination (Commission International de l'Éclairage – CIE) and is defined as a function of the tristimulus values *X*, *Y* and *Z*, expressed as:

$$X = k \int \Phi(\lambda)\bar{x}(\lambda)d\lambda \tag{3-39}$$

$$Y = k \int \Phi(\lambda)\bar{y}(\lambda)d\lambda \tag{3-40}$$

$$Z = k \int \Phi(\lambda)\bar{z}(\lambda)d\lambda \tag{3-41}$$

where $\Phi(\lambda)$ represents the spectral power of the stimulus and $\bar{x}(\lambda)$, $\bar{y}(\lambda)$ and $\bar{z}(\lambda)$ are the colour-matching functions of the 1931 CIE standard observer. As the spacing of the colours in this *XYZ* space is not perceptually uniform, CIE 1976 $L^*a^*b^*$ (CIELAB) colour space was defined according to the following transformations applied to the *XYZ* coordinates:

$$L^* = 116\left[f\left(\frac{Y}{Y_n}\right) - \frac{16}{116}\right] \tag{3-42}$$

$$a^* = 500\left[f\left(\frac{X}{X_n}\right) - f\left(\frac{Y}{Y_n}\right)\right] \quad (3\text{-}43)$$

$$b^* = 200\left[f\left(\frac{Y}{Y_n}\right) - f\left(\frac{Z}{Z_n}\right)\right] \quad (3\text{-}44)$$

with

$$f(w) = \begin{cases} w^{1/3} & \text{if } w > 0.008856 \\ 7.787w + \dfrac{16}{116} & otherwise \end{cases} \quad (3\text{-}45)$$

where a^* and b^* denote chromaticity, L^* denotes lightness and X_n, Y_n and Z_n are the tristimulus values of the reference white, usually chosen to be 0.9642, 1.0 and 0.8249, respectively.

Since an edge is essentially an abrupt difference between colour values of adjacent pixels, still according to the CIELAB standard, it can be represented as:

$$\Delta E^* = \sqrt{[(\Delta L^*)^2 + (\Delta a^*)^2 + (\Delta b^*)^2]} \quad (3\text{-}46)$$

where

$$\Delta L^* = L_1^* - L_0^* \quad (3\text{-}47)$$

$$\Delta a^* = a_1^* - a_0^* \quad (3\text{-}48)$$

$$\Delta b^* = b_1^* - b_0^* \quad (3\text{-}49)$$

and L_0^*, a_0^* and b_0^* represent the mean values of the background colour.

To get to this point, the values of the RGB colour space had to be converted to the XYZ [130,131] colour space. This was achieved in two steps. The initial step consisted on converting them to the

standard RGB – sRGB – colour space proposed by Stokes et al. [76]. This meant finding the values for R_s, G_s and B_s, which could done by executing the following calculations:

$$R' = R/255.0 \quad (3\text{-}50) \qquad G' = G/255.0 \quad (3\text{-}51) \qquad B' = B/255.0 \quad (3\text{-}52)$$

R_s was then computed according to:

$$R_S = \begin{cases} \dfrac{R'}{12.92} & \text{if } R' \leq 0.03928 \\ \left[\dfrac{R' + 0.055}{1.055}\right]^{2.4} & otherwise \end{cases} \quad (3\text{-}53)$$

and G_s and B_s were found in the same way.

After having these values, X, Y and Z could be reach by doing:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.4124 & 0.3576 & 0.1805 \\ 0.2126 & 0.7152 & 0.0722 \\ 0.0193 & 0.1192 & 0.9505 \end{bmatrix} \begin{bmatrix} R_S \\ G_S \\ B_S \end{bmatrix} \quad (3\text{-}54)$$

The image resulting from the application of the above described filters becomes a grey level image where the lesion is represented by bright pixels – whose colour values are more distant from the colour of the image background - and the background, itself, by dark ones, as can be seen in figure 3-19.

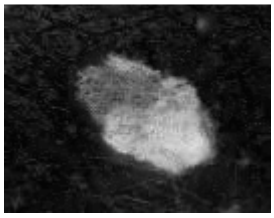


Figure 3-19- CIELAB filtered image

3.2.5.2 Smoothing of the greyscale image

The next step of the process consists of "smoothing" the image. This operation is done because it very important that we can stress the differences in intensity at the neighbourhood of the edges, as well as reduce these differences if the points being processed are part of the background or part of the lesion interior – we want the intensity values within the two segments of the image to be as uniform (without noise) as possible. The application of a Gaussian filter can do this job.

To be able to apply the filter it was necessary to define a suitable standard deviation - σ - value for the image background. This value was calculated by taking the median of the colours of every pixel on four small square areas on the image corners – assuming they were not occupied by the lesion – and applying the well known formula for calculating σ:

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})^2} \tag{3-55}$$

The resulting Gaussian filtered image was then created by applying the following function to the previous shown image:

$$I(x) = \frac{1}{\sqrt{2\pi}\sigma}\left(1 - e^{\left(-\frac{x^2}{2\sigma^2}\right)}\right) \tag{3-56}$$

and looked like this:

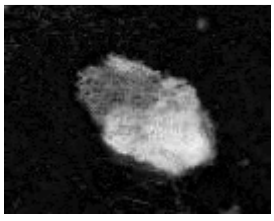


Figure 3-20 - Smoothed image

3.2.5.3 Thresholds

The following step in the process has the objective of defining, in a rough way, the position of the lesion within the image. With this in mind, the first task was finding three threshold values, T, T_1 and T_2. This is accomplished through the creation of an image pixels intensities histogram. T_2 is the value corresponding to the intensity level closer to 255 that has points within the image. T_1 is defined as a percentage - input as a parameter - of the value corresponding to T_2. The remaining threshold is calculated by doing $T = (T_1 + T_2)/2$.

The two thresholds T_1 and T_2 were then applied to the previously smoothed image. The result is shown in figure 3-21.

Figure 3-21 - Image resulting from applying the thresholds T_1 and T_2 to the smoothed image

3.2.5.4 Edge detection

The next action was applying a previously described Sobel filter to the smoothed grey level image. The result of this step was the image represented in figure 3-22.

Figure 3-22 - Output of the Sobel filter applied to the smoothed grey level image

Combining these two last images, it is possible to select every point that has a maximum gradient value and lies within the set of points selected by applying the thresholds T_1 and T_2. From this we get an image – figure 3-23 - with a rough edge but already very near the desired objective.

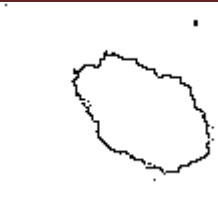


Figure 3-23 - Edge resulting from the combination of the outputs from the Sobel filter and the double thresholds

The next step in the method consists on applying a threshold *T*. This gives us the output represented by figure 3-24.

Figure 3-24 - Output from applying the *T* threshold

Since this threshold value is between the previous values for T_1 and T_2, the limits of this image's areas must be enclosed within the edge represented in figure 3-23. Working on the image represented by figure 3-24 and verifying which of the previous edge points are closer to the points that limit its areas, results in an improved - thinner, although not necessarily closed - edge, as can be seen below.

3.2.5.5 Closing and thinning

It is now imperative that this edge be closed. To achieve this goal, the points on the already detected edge were inserted into an array and processed, one by one.

Figure 3-25 - Thinned - not yet closed - edge

As a first approach to making the edge as close to one pixel wide as possible, the non allowable configurations described by Helterbrand, J. D. [77] and represented in figure 3-26 were corrected. Point sets found with the depicted configurations will have the pixels marked with a square, removed.

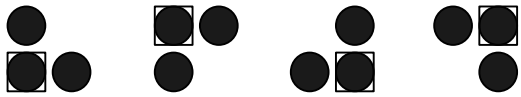


Figure 3-26 - Non allowable pixel configurations

For every point on the array, every other element of the data structure was searched in order to find the nearest point within the edge and in the case where several pixels were adjacent to the one being worked upon; the selected pixel was the one with the highest gradient value. All the other pixels in the neighbourhood were deleted. This originated a thinner edge, just like the one in figure 3-27.

Figure 3-27 - Thinner edge (still not closed)

The two points were then connected by a line segment. Once closed, the edge points were then subject to a rational Gaussian curve modelling [78,132] as a way to smooth and accommodate the drawn line segments to the real edge. A rational Gaussian curve – RaG – with control points $\{V_i : i = 1, \dots, n\}$ is defined by:

$$P(u) = \sum_{i=1}^{n} V_i g_i(u) \qquad u \in [0,1] \tag{3-57}$$

where

$$g_i(u) = \frac{W_i G_i(u)}{\sum_{j=1}^{n} W_j G_j(u)} \quad (3\text{-}58)$$

is the *i*th basis function of the curve, W_i is the weight of the *i*th control point, and

$$G_i(u) = e^{[-(u-u_i)^2/2\sigma^2]} \quad (3\text{-}59)$$

is a Gaussian function of height 1 and standard deviation σ_i, centred at u_i.

The result from all this process, although not always a perfect one pixel wide edge, is nevertheless thin enough to be used for the purpose of this work. The final result can be seen in figure 3-28, with the finally detected edge superimposed on the initial image.

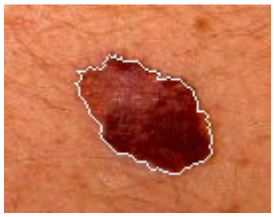


Figure 3-28 - Detected edge

3.2.6 Size

If the images to work with are taken with the same resolution and at a uniform distance from the focal point of the camera – which is not difficult to achieve - then, this feature can be analysed as a relative value and the number of points within the previously detected edge is a good approximation to the real number.

3.2.7 Diameter

For finding values for this as well as all the remaining features, another program was developed. Its main panel is represented in figure 3-29.

To implement it, a probable centre point was calculated. This was achieved by finding the crossing point coordinates for two extreme orthogonal diameters. Knowing this point's coordinates (x_c, y_c) it is

now possible to calculate the Euclidean distance between it and every point on the mole's edge. This was given by:

$$r_i = \sqrt{(x_i - x_c)^2 + (y_i - y_c)^2} \tag{3-60}$$

The average radius - distance from the centre point to the edge – is then calculated by:

$$D_{avg} = 2r_{avg} = \frac{2}{N}\sum_{i=1}^{N} r_i \tag{3-61}$$

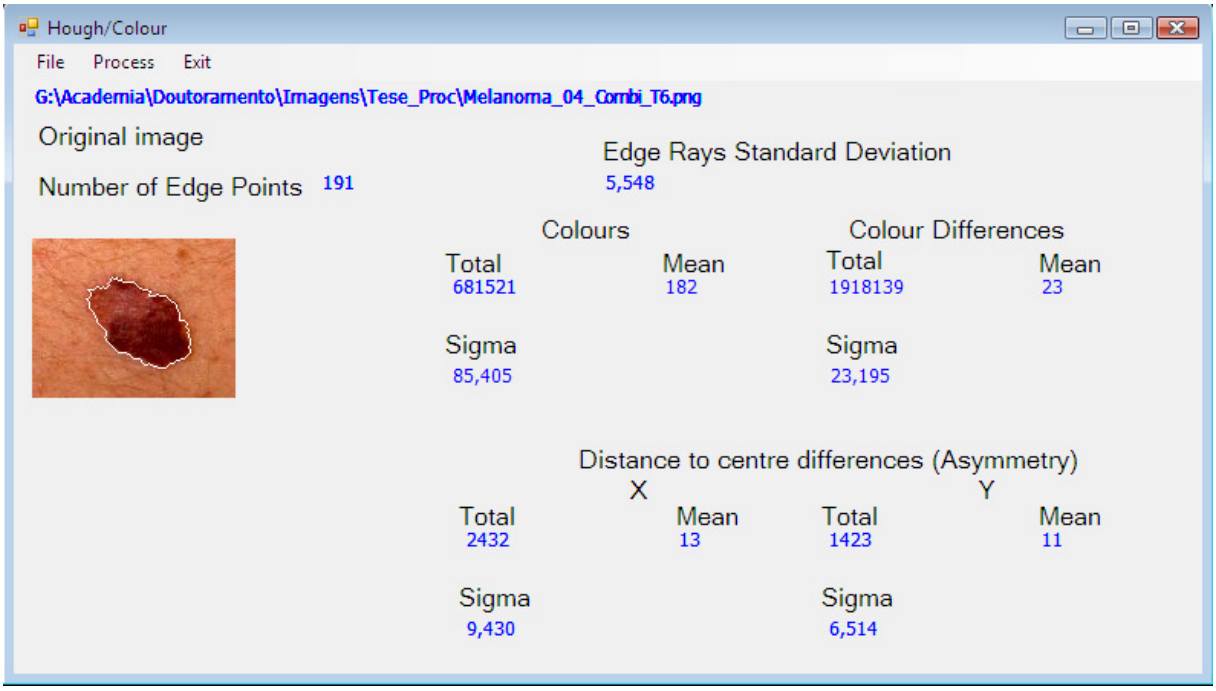


Figure 3-29 – Asymmetry - Colour program's panel

3.2.8 Jaggedness

Asymmetry was calculated taking into account two extracted features: Number of changes in direction of the tumour's edge and standard deviation of the edge radius.

The first value can be easily calculated by moving along the edge line.To calculate the second, one must first determine a probable centre point for the mole. To achieve this, the maximum dimensions of the lesion along two orthogonal directions were computed. The middle point, defined by half the difference between the maximum and minimum coordinates along the two directions, was accepted as a probable centre point. With this value it was possible to define the variations of the distance between the probable centre point and every point on the lesion's edge.

Knowing the above values, it is now possible to calculate the standard deviation between the values of r_i. This might, however, not be an accurate measure of jaggedness. In fact, if there were many changes in the direction of the edge, the various resultant edge segments could compensate for each other, resulting in a false notion of regularity in the size of the diameters. To account for this problem, the number of changes in the mole's edge direction was also counted.

3.2.9 Colour detection

More important for the diagnosis of malignant skin tumours than their colour itself, is the number of colours contained within the mole's edge. So, this feature extraction routine is concentrated on, not only calculating the standard deviation of the mole's pixels colour values but also on measuring the standard deviation of the colour differences between adjacent pixels – colour variations. The values for Minimum Colour, Maximum Colour and Total Colours due to their volatility – dependent on several external factors – were only used to calculate derived values, as independent as possible.

3.2.10 Calculated values

Since some of the extracted feature values seem to be closely related, ratios between their values were also calculated, namely:

$$TCD\_TC = \frac{Total_{Colour\ Differences}}{Total_{Colours}} \tag{3-62}$$

which will account for the normalization of the changes in colour;

$$DC\_EP = \frac{Direction\ Changes_{edge}}{Nr.of\ points\ on\ the\ edge} \tag{3-63}$$

which, along with:

$$EP\_LP = \frac{Nr.of\ points\ on\ the\ edge}{Nr.of\ points\ within\ the\ lesion\ boundary} \quad (3\text{-}64)$$

will represent how smooth the edge line is.

All the processed images, with detected edges, as well as tables with both all the directly extracted features and the calculated ones can be seen on Appendix E.

As can be seen, edges are well defined and, in most of the cases, 1 pixel wide. Images where this is not the case have very few duplicated pixels and they did not have a significant influence on the extracted features. Although the whole process is rather complex, results – a clear edge – are achieved very fast.

4 Training and testing the system

The previously developed programs – addressing all the relevant features – were applied to a subset of the available photos and the results were evaluated according to the previous image classification – done by a medical expert. All the systems were trained using the same subset of images and were afterwards tested also with the tests subset. Training subsets with, 50%, 60%, 70% and 80% of the whole amount of available images were used. The remaining images were used for testing the system's performance.

Prior probabilities for the system are calculated according to the distribution of malignant and benign lesions within the samples file. This should be rather acceptable since in cases like this, it is better to get false positives than letting a malignant lesion undetected.

Both training and testing were executed with the help of an open source software package called Weka© version 3.4.12, developed by the New Zealand University of Waikato.

4.1 Features evaluation

The various features extracted from the sample images were evaluated as to their relevance for the job, according to the criteria pointed out by Dermatology specialists as the most significant for classification effects: Size, Colour and Jaggedness – see point 8.5.3 - Appendix E. Although the total colour feature was rather well classified, since none of the sample images has been subject to a colour correction process, it has been decided not to use that feature as well as every other dependent on absolute colour value. Instead, and according to the medical experts opinion [84,85] ratios between features were used. This allowed for an almost complete independence between results and image formats or illumination conditions. Since accuracy is for this work, more important than processing time, all other extracted features were used.

Several other statistical data were calculated by the package, namely:

1. **Correctly and Incorrectly classified instances**
   Number of images correctly and incorrectly classified, both during the training phase and the test phase;
2. **Confusion matrix**
   Represents the relations between real and predicted values. As an example, the next matrix corresponds to the results of the application of the naïve Bayes algorithm to the samples file with 50% of the images used for training and the other 50% for testing.

Table 4-1 - Confusion matrix for naïve Bayes

	Predicted Malignant	**Predicted Benign**
True Malignant	35	9
True Benign	7	17

3. **Kappa statistic** [80,81,82]
   This is a measure of the differences between the classification result values agreements obtained through the use of the classifier and those that would be expected by the simple use

of chance. A standardized value, it will lie between -1 and 1 with 1 representing a perfect agreement between the two methods, 0 is exactly what would be expected by chance, and negative values mean an agreement worse than chance, that is, a potential systematic disagreement between the methods. This value can then be represented by:

$$K = \frac{P_{CA} - P_C}{1 - P_C} \tag{4-1}$$

where P_{CA} is the number of classification agreements and P_C is the proportion of agreements expected by chance.

4. **Kononenko and Bratko's (K&B) Information Score** [79,83]

With a limited number of samples, classification tasks usually result in some errors, false positives or false negatives – in this case, malignant spots being classified as benign or vice versa. As the number of samples increases, the relation between well classified samples and wrongly classified ones becomes smaller and the apparent success rate approaches 100%. This, of course, will not allow for an effective measure of the classifier's precision. To try to solve this problem, in 1991, Kononenko & Bratko introduced this Information Score, which takes into consideration the values of prior probabilities. Since the level of information – Information Score – associated with a correct positive classification is:

$$-\log_2 P(M) \tag{4-2}$$

where, $P(M)$ is the prior probability of the **Malignant** class. The Information Score can then be used to weigh all the classifier's results. Since we have two classes, **Malignant** and **Benign**, the Information Scores can be represented by the following matrix:

Table 4-2 - Information Score matrix

	Predicted Malignant	**Predicted Benign**
True Malignant	$-\log_2 P(Malignant)$	$-\log_2\big(1 - P(Benign)\big)$
True Benign	$-\log_2\big(1 - P(Malignant)\big)$	$-\log_2 P(Benign)$

Multiplying each element of the confusion matrix by the corresponding element of the information score matrix and dividing by the total number of samples, will result in another, scaled, matrix. Subtracting the values relative to the classifier's predictions, we get the number of bits of information associated with each sample.

5. Mean absolute error

This is the weighted average of the absolute errors, with the relative frequencies as the weight factors and can be represented by:

$$E = \frac{1}{n}\sum_{i=1}^{n}|P_i - T_i| \tag{4-3}$$

With P_i - representing the predicted value - and T_i - the target value for sample i.

6. **Root mean squared error**

A measure of the differences between values predicted by the classifier and the actual sample class, represented as:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}|P_i - T_i|^2} \tag{4-4}$$

7. Relative absolute error

Obtained dividing the mean absolute error by the corresponding error of the ZeroR classifier on the data (i.e. the classifier predicting the prior probabilities of the classes observed in the data);

8. Root relative squared error

Results of the division of the root mean squared error by the corresponding error of the ZeroR classifier on the data;

4.2 Results

The results obtained by the application of the various algorithms on the input file are represented below, grouped by percentage of the file used for training the system. As can be concluded by the analysis of the following tables, and as expected, the increase in the number of training samples drives much accurate classification results. With 50% of the images used for training, the best results achieved barely surpass the 82% and result from the application of a Support Vector Machine classifier. Although already rather good when compared to the 60% average achieved by human experts, these results are very much improved when the number of training samples reaches 80% of the total image set. The correctly classified images are now, almost 93% of the whole group of samples kept for testing the process. The best classifiers are then, a Tree Augmented Naïve Bayes or a Multilayer Perceptron.

As important as the well classified samples – if not more important – are the values registered for incorrectly classified images. These go from almost 18% of the classified images for the Support Vector Machine, to around a mere 7% for the two best performers.

These are very promising results when compared to other pieces of software/equipment available in the market or being developed by academic institutions.

Some of these systems and their results have already been referenced in section 1.4 - Work on the field, nevertheless, so that these system achievments can be compared with other realities, some other systems - with some similarities to the object of this research - and their related results are introduced below.

- Celebi et al. [154] in their 2007 paper "*A methodological approach to the classification of dermoscopy images*" claim to have used 564 dermoscopy images – and achieved a true positives rate – sensitivity - of 93,3% by applying a Support Vector Machine classifier. In this paper they also included a table with results for several other recent studies using several segmentation methods and classifiers like; kNN, Artificial Neural Networks and Logistic regression, where the values achieved range from values around 73% to 93%. Two other results are shown with 100% true positives but, in one of these cases, the classifier is not

reported and, in the other, the number of images - 40 – is, in the author's opinion, too small for the results to be considered reliable.

- Messadi, Bessaid and Taleb-Ahmed [155], in 2009, published an article within the Journal of Medical Engineering & Technology, where they report values between 65% and 74% of correct classifications performed by a multi-layer perceptron, results they claim to be "*comparable with the detection rates of very experienced dermatologists*".

- Marozas & Jurkonis [156] during the 12th International Conference on Biomedical Engineering that took place in Lithuania October, 2008 presented a paper where they analysed several available skin tumour classification systems and concluded that: "*most of those software solutions have some major disadvantages*".

- Sigurdsson et al., in 2004 [157], published an article stating that they had developed a system with which "*skin lesion classification based on in vitro Raman spectroscopy is approached using a nonlinear neural network classifier*". According to the authors, "*the classification performance for the present data set, involving 222 cases and five lesion types, was 80.5%±5.3% correct classification of malignant melanoma, which is similar to that of trained dermatologists based on visual inspection*"

4.2.1 Confusion matrices

Within this section are the confusion matrices that result from the application of the tested classifiers – Fuzzy k-NN first, followed by all other classifiers, with test sets that were formed by 50%, 60%, 70% and 80% of the total set of available images. Confusion matrices represent the number of correctly and incorrectly classified lesions. An incorrect classification may result in what is known as a false positive – when a benign lesion is wrongly classified as malignant - or a false negative – a malignant tumour classified as benign.

A good classification system should reduce these values to as close to zero as possible. It becomes clear, from the analysis of the following confusion matrices, that results become more accurate – as would be expected - as the number of images used to train the system increases. Most of all, in order to become a reliable method, the number of malignant lesions classified as benign must be minimized. It can be seen that this happens for a training set of 80% of all the available images and k-NN Classifier and Multilayer Perceptron.

Occurrences of 1 single false negative were also recorded with the k-NN classifier and training sets formed by as few as 60% and 70% of the whole group of photos processed. Another occurrence of a single false negative resulted from the application of the Tree Augmented Naïve Bayes classifier with a training set of 80% of the images. Although not perfect, these latter results show that with a relatively small number of training images these classifiers are still able to achieve a considerable high level of accuracy.

4.2.1.1 **Fuzzy k-NN**

Table 4-3 - Confusion matrix for Fuzzy k-Nearest Neighbour method

Fuzzy k-NN (k = 9)		**Predicted**	
		Malignant	**Benign**
True	**True Malignant**	85	7
	True Benign	13	31

4.2.1.2 Training set - 50%

Table 4-4 - Confusion matrices for a training set of 50% of the total samples

Naïve Bayes		Predicted	
		Malignant	**Benign**
True	**True Malignant**	29	15
	True Benign	0	24

Tree Augmented Naïve Bayes (TAN)		Predicted	
		Malignant	**Benign**
True	**True Malignant**	35	9
	True Benign	7	17

k-Nearest Neighbours (k = 9)		Predicted	
		Malignant	**Benign**
True	**True Malignant**	39	5
	True Benign	8	16

Support Vector Machine		Predicted	
		Malignant	**Benign**
True	**True Malignant**	41	3
	True Benign	9	15

Multilayer Perceptron		Predicted	
		Malignant	**Benign**
True	**True Malignant**	35	9
	True Benign	5	19

4.2.1.3 Training set - 60%

Table 4-5 - Confusion matrices for a training set of 60% of the total samples

Naïve Bayes		Predicted	
		Malignant	**Benign**
True	**True Malignant**	24	11
	True Benign	0	20

Tree Augmented Naïve Bayes (TAN)		Predicted	
		Malignant	**Benign**
True	**True Malignant**	33	2
	True Benign	6	14

k-Nearest Neighbours (k = 9)		Predicted	
		Malignant	**Benign**
True	**True Malignant**	34	1
	True Benign	5	15

Support Vector Machine		Predicted	
		Malignant	**Benign**
True	**True Malignant**	33	3
	True Benign	9	10

Multilayer Perceptron		Predicted	
		Malignant	**Benign**
True	**True Malignant**	28	7
	True Benign	2	18

4.2.1.4 Training set - 70%

Table 4-6 - Confusion matrices for a training set of 70% of the total samples

Naïve Bayes		Predicted	
		Malignant	**Benign**
True	**True Malignant**	18	17
	True Benign	0	16

Tree Augmented Naïve Bayes (TAN)		Predicted	
		Malignant	**Benign**
True	**True Malignant**	22	3
	True Benign	4	12

k-Nearest Neighbours (k = 9)		Predicted	
		Malignant	**Benign**
True	**True Malignant**	24	1
	True Benign	6	10

Support Vector Machine		Predicted	
		Malignant	**Benign**
True	**True Malignant**	26	2
	True Benign	5	8

Multilayer Perceptron		Predicted	
		Malignant	**Benign**
True	**True Malignant**	22	3
	True Benign	3	13

4.2.1.5 Training set - 80%

Table 4-7 - Confusion matrices for a training set of 80% of the total samples

Naïve Bayes		Predicted	
		Malignant	**Benign**
True	**True Malignant**	14	4
	True Benign	0	10

Tree Augmented Naïve Bayes (TAN)		Predicted	
		Malignant	**Benign**
True	**True Malignant**	17	1
	True Benign	1	9

k-Nearest Neighbours (k = 9)		Predicted	
		Malignant	**Benign**
True	**True Malignant**	18	0
	True Benign	3	7

Support Vector Machine		Predicted	
		Malignant	**Benign**
True	**True Malignant**	17	2
	True Benign	4	5

Multilayer Perceptron		Predicted	
		Malignant	**Benign**
True	**True Malignant**	18	0
	True Benign	2	8

4.2.2 Other results

The next set of tables shows the values calculated for all the statistical ratios described in section 4.1. This has been done for all the used classifiers and for every test set.

4.2.2.1 **Fuzzy k-Nearest Neighbour (k=9)**

Table 4-8 – Fuzzy k-Nearest Neighbour

Nr. of test samples = 136	Classified				Kappa	K&B IS (bits/instance)	Mean Absolute Error	Root mean squared error	Relative absolute error	Root relative squared error
	Correctly		Incorrectly							
Fuzzy k-Nearest Neighbour (k=9)	116	85.2941 %	20	14.7059 %	0.6516	0.4478	0.2268	0.3354	51.6766 %	71.654 %

4.2.2.2 Training set - 50%

Table 4-9 - Other results for a training set with 50% of the total samples

Nr. of test samples = 68	Classified				Kappa	K&B IS (bits/instance)	Mean Absolute Error	Root mean squared error	Relative absolute error	Root relative squared error
	Correctly		Incorrectly							
Naïve Bayes	54	79.4118 %	14	20.5882 %	0.602	0.4719	0.2125	0.4476	48.1727 %	93.0941 %
Tree Augmented Naïve Bayes (TAN)	52	76.4706 %	16	23.5294 %	0.4944	0.4382	0.2233	0.3945	50.6068 %	82.0471 %
k-Nearest Neighbours (k = 9)	55	80.8824 %	13	19.1176 %	0.5692	0.418	0.2557	0.342	57.9587 %	71.1383 %
Support Vector Machine	**56**	**82.3529 %**	**12**	**17.6471 %**	**0.5904**	**0.5487**	**0.1765**	**0.4201**	**40.0000 %**	**87.3704 %**
Multilayer Perceptron	54	79.4118 %	14	20.5882 %	0.5657	0.4966	0.1995	0.3904	45.2140 %	81.1934 %

4.2.2.3 Training set - 60%

Table 4-10 - Other results for a training set with 60% of the total samples

Nr. of test samples = 55	Classified				Kappa	K&B IS (bits/instance)	Mean Absolute Error	Root mean squared error	Relative absolute error	Root relative squared error
	Correctly		Incorrectly							
Naïve Bayes	44	80.0000 %	11	20.0000 %	0.6134	0.5481	0.1770	0.4004	39.7011 %	82.5449 %
Tree Augmented Naïve Bayes (TAN)	47	85.4545 %	8	14.5455 %	0.6716	0.5917	0.1822	0.3008	40.8610 %	62.0051 %
k-Nearest Neighbours (k = 9)	**49**	**89.0909 %**	**6**	**10.9091 %**	**0.7537**	**0.5127**	**0.225**	**0.295**	**50.4723 %**	**60.8087 %**
Support Vector Machine	43	78.1818 %	12	21.8182 %	0.4787	0.4497	0.2182	0.4671	49.3314 %	98.0055 %
Multilayer Perceptron	46	83.6364 %	9	16.3636 %	0.6644	0.5961	0.1615	0.3361	36.2209 %	69.2915 %

4.2.2.4 Training set - 70%

Table 4-11 - Other results for a training set with 70% of the total samples

Nr. of test samples = 41	Classified				Kappa	K&B IS (bits/instance)	Mean Absolute Error	Root mean squared error	Relative absolute error	Root relative squared error
	Correctly		Incorrectly							
Naïve Bayes	34	82.9268 %	7	17.0732 %	0.6674	0.6344	0.1563	0.3791	34.2866 %	76.3847 %
Tree Augmented Naïve Bayes (TAN)	34	82.9268 %	7	17.0732 %	0.6372	0.5447	0.2122	0.3390	46.5509 %	68.309 %
k-Nearest Neighbours (k = 9)	34	82.9268 %	7	17.0732 %	0.6199	0.4871	0.2472	0.3334	54.2267 %	67.185 %
Support Vector Machine	34	82.9268 %	7	17.0732 %	0.5798	0.5299	0.1707	0.4132	39.0006 %	88.7617 %
Multilayer Perceptron	**35**	**85.3659 %**	**6**	**14.6341 %**	**0.6925**	**0.6159**	**0.1691**	**0.3337**	**37.1044 %**	**67.2514 %**

4.2.2.5 Training set - 80%

Table 4-12 - Other results for a training set with 80% of the total samples

Nr. of test samples = 28	Classified				Kappa	K&B IS (bits/instance)	Mean Absolute Error	Root mean squared error	Relative absolute error	Root relative squared error
	Correctly		Incorrectly							
Naïve Bayes	24	85.7143 %	4	14.2857 %	0.7143	0.6758	0.1172	0.3158	26.1508 %	65.6990 %
Tree Augmented Naïve Bayes (TAN)	**26**	**92.8571 %**	**2**	**7.1429 %**	**0.8444**	**0.6236**	**0.1542**	**0.2746**	**34.4100 %**	**57.1242 %**
k-Nearest Neighbours (k = 9)	25	89.2857 %	3	10.7143 %	0.7500	0.4920	0.2307	0.3073	51.4924 %	63.9253 %
Support Vector Machine	22	78.5714 %	6	21.4286 %	0.4783	0.4382	0.2143	0.4629	48.8889 %	99.1112 %
Multilayer Perceptron	**26**	**92.8571 %**	**2**	**7.1429 %**	**0.8372**	**0.6618**	**0.1372**	**0.2809**	**30.6199 %**	**58.4236 %**

4.2.2.6 **Analysis**

The above tables show correct classifications values between over 82% to almost 93%. More than this they present results for two of the classifiers – Tree Augmented Naïve Bayes and Multilayer Perceptron – that are just above the 7% mark.

For Kappa statistic – which as said before is a ratio that represents the differences between the classification result values agreements obtained through the use of the classifier and those that would be expected by the simple use of chance – values span from about 0.5 to very near 0.85. These latter values, particularly, very near the deterministic value of 1 for the ratio, show us that the system has already a good level of reliability.

Values for the K&B information score from around 0.4 to over 0.6 bits per instance with the Tree Augmented Naïve Bayes and near 0.7 bits per instance with the Multilayer Perceptron are in line with the previous ratios and showed us that these two classifiers become best performers as the number of test images grows.

The various types of calculated errors, either absolute or relative to the iso-probable, chance, classifier – ZeroR - also present values that are very encouraging, namely in what the previously referred two classifiers - Tree Augmented Naïve Bayes and Multilayer Perceptron are concerned.

5 Conclusions

The number of new cases of melanoma every year [140] together with the percentage of total remissions that can be achieved when the lesions are detected in their first stages of evolution turns early detection a simple case of common sense.

Dermatology centres are very specialized work environments not always available in small clinical institutions. Both these conditions lead to the necessity of developing an easy to use and highly trustable early diagnosis system. Up to now, all the available systems rely on rather sophisticated and expensive equipment, not directed or even accessible to the general public. This study intended to show that it was possible, with some common utilization hardware – a simple digital camera – and some well structured software, to reach results that would increase the possibilities of accurate detection of malignant skin lesions and, consequently, increase the possibilities of recovery from skin cancer, thus, also increasing life expectancy of the people affected by this disease. Regardless of the cost associated with the available systems, most of them are nothing but simple image storage solutions. Those few that implement classification methods, reach results that are much less than satisfactory.

In conversations with the staff of the Dermatology department of a Portuguese reference Hospital - headed by Dr. Campos Lopes at the time - it has been referred to the author that the best results achieved by all the systems tested by the service personnel were only able to achieve around 50% of correct classifications.

These considerations made it obvious for the author that, along the research process, various difficulties would arise. This became a fact, namely in what image capturing conditions were concerned. These heterogeneous environmental conditions create several problems to the feature extraction system, particularly in the detection of edges, which is, in fact, the basis for the classification process.

It became very clear, from the tests made along the research that no edge detection or segmentation algorithm on its own would be able to guarantee a reliable feature extraction process. The combined method, instead, although not perfect, allowed for an edge detection that was coherent and good enough for the objectives of this work.

This system only allows for the processing of one lesion per image, although with better results than any other system available to this moment. To be able to detect several lesions – and classify them correctly – within one single image is a theme for future research.

The features used to classify the images were selected and extracted or calculated based on a totally innovative concept – mostly relations, rather than absolute values – in order to answer the main characteristics identified by the experts: Asymmetry, Border, Colour – more specifically, changes in colour within the lesion's area - and Diameter. These are, of course, parameters that were consciously identified. Nevertheless, it is not exactly clear how a classification program uses them. In fact, since their definitions are rather subjective, it was not easy to extract and calculate features that might correctly represent the above characteristics. At first, 26 features were extracted / calculated, of which, 23 were presented to every classification algorithm. Given the sensibility of the matter, even though a smaller number of features could have been selected, it was decided to sacrifice execution time vs. classification precision, and so, work with all of them.

Since during the research process, it was impossible to find samples collected in different moments in time, the growth parameter could not have been taken into account. If and when that becomes possible, I believe it will also be possible to enhance, even more, the classification results. In fact, the integration of all the work already developed with a database containing not only images of the same lesions, captured in various moments in time, but also personal characteristics like patient's age and gender, will surely allow for more accurate classification results.

With all these issues addressed, the classification results, seen on the previous pages, were quite interesting. They have shown us that in a situation where the training set is relatively small (50% of the total samples) the best performing algorithm is the Support Vector Machine, although with rather high levels of error.

Table 5-1 - Best performer algorithm with 50% samples training set

Support Vector Machine		Kappa statistic	0.5904
Nr. of training set samples	68	K&B IS (bits/instance)	0.5487
Correctly classified	56	Mean Absolute Error	0.1765
	82.3529 %	Root mean squared error	0.4201
Incorrectly classified	12	Relative absolute error	40.0000 %
	17.6471 %	Root relative squared error	87.3704 %

As the number of training samples grows the algorithms responses change along with it. Above the 60% of the total number of samples case, the best performances are achieved by the Tree Augmented Naïve Bayes (TAN) algorithm and the Multilayer Perceptron, almost side by side. With more than 92% of correct test responses – with 80% of the whole images set used for training - it performs rather satisfactorily. For detailed classification results, please refer to section 8.6 – Appendix F.

Table 5-2 - TAN performance with 80% samples training set

Tree Augmented Naïve Bayes (TAN)		Kappa statistic	0.8444
Nr. of training set samples	108	K&B IS (bits/instance)	0.6236
Correctly classified	26	Mean Absolute Error	0.1542
	92.8571 %	Root mean squared error	0.2746
Incorrectly classified	2	Relative absolute error	34.4100 %
	7.1429 %	Root relative squared error	57.1242 %

Table 5-3 - Multilayer Perceptron performance with 80% samples training set

Multilayer Perceptron		Kappa statistic	0.8372
Nr. of training set samples	108	K&B IS (bits/instance)	0.6618
Correctly classified	26	Mean Absolute Error	0.1372
	92.8571 %	Root mean squared error	0.2809
Incorrectly classified	2	Relative absolute error	30.6199 %
	7.1429 %	Root relative squared error	58.4236 %

From all the above calculated ratios, it was assumed acceptable to conclude that both the Tree Augmented Naïve Bayes (TAN) and the Multilayer Perceptron, if fed with more than, around, 100 images for training, originate rather accurate classification results.

In this case, not only the values for the number of correctly classified images are above 90% but also every other ratio assumes very significant values. A particular and rather relevant ratio is the kappa statistic whose value – 0.844 and 0.837 - is very near 1, representing an almost total agreement between observations and consequently a very good level of accuracy of the classification method.

This work is a totally novel combination of already proven algorithms – implemented in a multistage process - and new features and feature relations. Although with a somewhat still limited scope – one lesion per image – and even with the limited amount of test images that were able to be gathered – all previously classified - the results achieved by this arrangement of well known segmentation algorithms with the novel feature selection and extraction processes implemented, together with several well known classification algorithms, showed this seems to be the right path to achieving a reliable early screening system. If and when, to all these data, values for age, gender and evolution might be used as classification features, the results will, no doubt, become even more accurate, allowing for an improvement in the survival rates of skin cancer patients.

The results depicted herein are, to the authors' knowledge, better than what can, up to now, be achieved by any system available. They are also largely better than the 60% of correct classifications that human experts are capable of, without any laboratory analysis.

All these previous considerations lead the author to believe this is one of the most accurate skin lesions classification systems available both in the market and within the academic environment. This conclusion stands, although the set of test images used here, being limited in numbers and previously classified might be prone to some kind of bias in the process of classification. To better evaluate the system's performance; new non-previously classified images will be presented to it as they become available.

6 Future work

It is the author's belief that the system still needs, nevertheless, to be subject to more extensive tests. Trying to check its results against non-previously classified images is one of the tasks that should be implemented within the near future, in order to be able to evaluate if it is able to perform in such an accurate way with any other collected image.

Anyway, the precision of the results achieved so far, encourage the author to, under the scope of future work, turn the focus, not only to the improvement of the feature extraction process but also to the possible utilization of new combinations of features.

Another area of future work is related to the possibility of treating images with more than one lesion. This imposes different challenges from the ones addressed in this work so far. An image of a portion on human skin with various lesions will have to be processed in an even more sophisticated way. Lesions must be automatically individualized and this fact will require new algorithms to be inserted within the combined edge detection method used here.

For being able to follow the evolution of the moles – one of the most important classification factors not yet considered – as well as for taking into consideration the age and gender of the patients – two other statistically relevant characteristics - it will be important that a database be integrated with the rest of the already developed system. This database tables should include both values for one lesion in various moments in time and personal data about the patient by the time the image was captured.

The possible insertion of this piece of software in a portable hardware device can lead to such autonomy that it will allow for everyone, no matter how far they are from a central hospital, to be able to early diagnose cases of skin cancer and, in that way, make it possible to treat them in due time, significantly increasing the life expectancy of the people affected by such a problematic disease.

7 References

1. http://www.skincancer.org/skincancer-facts.php
2. http://www.skincancer.org/self_exam/look_for.php
3. Theodoridis, S. and Koutroumbas, K. - Pattern Recognition 2nd Ed. – Elsevier Academic Press, 2003
4. Jähne, B. – Digital Image Processing, 5th Ed. – Springer-Verlag, 2002
5. Haykin, S. – Neural Networks, A Comprehensive Foundation, 2nd Ed. – Prentice Hall, 1999
6. Bishop, C.M. – Neural Networks for Pattern Recognition – Oxford University Press, 1995-2003
7. Jordan, M.I. and Sejnowsky, T.J. – Graphical Models, Foundations of Neural Computation – The MIT Press, 2001
8. Schwarzer, G., Vach. W and Schumacher, M. - Misuses of ANN for diagnostic classification in oncology – Statistics in Medicine 2000; 19: 541-561
9. Egmont-Petersen, M., Ridder, D. and Handels, H - Image processing with neural networks – A review – The Journal of the Pattern Recognition Society: 35 (2002) 2279-2301
10. Lampinen, J., Laakson, J. and Oja, E – Neural Network Systems, Techniques and Applications in Pattern Recognition – Helsinki University of Technology, Finland – http://zeus.hut.fi/publications/ps/b1_nnsystems.ps
11. Neal, R. M. - Probabilistic Inference Using Markov Chain Monte Carlo Methods – University of Toronto - 1993
12. Rajpoot, K., Rajpoot, N. and Turner, M. – Hyperspectral Colon Tissue Cell Classification – De Montfort University, Leicester, UK
13. Aubury, M. and Luk, W. – Binomial Filters – Journal of VLSI Signal Processing, i, 1-8, Kluwer Academic Publishers – 1995
14. Markovitch, S and Rosenstein, D. - Feature Generation Using General Constructor Functions – Machine Learning, 49, 59 – 98 – 2002

15. Larsen, J. - Design of Neural Network Filters – PhD. Thesis – Technical University of Denmark – 1996
16. Heath, M., Sarkar, S., Sanocki, T., and Bowyer, K. - Comparison of edge detectors: a methodology and initial study - Computer Vision and Pattern Recognition '96, San Francisco, 1996.
17. Heath, M., Sarkar, S., Sanocki, T., and Bowyer, K. - A Robust Visual Method for Assessing the Relative Performance of Edge-Detection Algorithms - IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 19, No. 12, pp. 1338-1359, December 1997.
18. Hu, M.K. - Visual pattern recognition by moment invariants - IEEE Trans. Inform. Theory, vol. IT-8, pp. 179--187, 1962.
19. Friedman, R., Geiger, D. and Goldszmidt, M. – Bayesian Network Classifiers – Kluwer Academic Publishers, Boston – 1997
20. Heckerman, D., Geiger, D. and Chickering, D.M. – Learning Bayesian Networks: The Combination of Knowledge and Statistical Data – Microsoft Research - 1995
21. Grossman, D. and Domingos, P. - Learning Bayesian Network Classifiers by Maximizing Conditional Likelihood - University of Washington, Seattle – 2004
22. Cheng, J. and Greiner, R. - Learning Bayesian Belief Network Classifiers: Algorithms and System – University of Alberta, Canada – 1999
23. Su, J. and Zhang, H. - Full Bayesian Network Classifiers – University of New Brunswick, Canada – 2006
24. Langley, P., Iba, W. and Thompson, K. - An analysis of Bayesian classifiers – NASA Ames Research Center – 1992
25. Samengo, I. - Estimating probabilities from experimental frequencies – Physical Review E, Volume 65 – 2002
26. Russell, S. and Norvig, P. – Artificial Intelligence: A Modern Approach – Pearson Education – 2003
27. Rish, I. - An empirical study of the naive Bayes classifier – IBM TJ Watson Research Center
28. Jain, A.K., Duin, R.P.W. and Mao, Jianchang - Statistical Pattern Recognition: A review – IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 22, No.1, January 2000
29. Rish, I., Hellerstein, J. and Thathachar, J. - An analysis of data characteristics that affect naive Bayes - IBM TJ Watson Research Center

30. Domingos, P. and Pazzani, M - On the Optimality of the Simple Bayesian Classifier under Zero-One Loss – University of California – 1997
31. Jing, Y, Pavlovic, V. and Rehg, J - Boosted Bayesian Network Classifiers – Georgia Institute of Technology and Rutgers University
32. Diao, L., Hu, K., Lu, Y. and Shi, C. - A Method to Boost Naïve Bayesian Classifiers – Tsinghua University, Beijing, China – 2002
33. Elkan, C. - Boosting And Naive Bayesian Learning – University of California – 1997
34. Ridgeway, G., Madigan, D., Richardson, T. and O'Kane, J. - Interpretable Boosted Naïve Bayes Classification – University of Washington – 1998
35. Efron, B. – Bootstrap methods: Another look at the jackknife – The Annals of Statistics, 1979, Vol. 7, No. 1, 1-26
36. Miller, R. – The Jackknife – A Review – Biometrika, Apr. 1974, Vol. 61, Issue 1, 1-15
37. Lowd, D and Domingos, P. - Naive Bayes Models for Probability Estimation – University of Washington
38. Jiang, L., Zhang, H., Cai, Z and Su, J. - Learning Tree Augmented Naive Bayes for Ranking – China University of Geosciences and University of New Brunswick
39. Cerquides, J and Màntaras, R. L. - Maximum a Posteriori Tree Augmented Naive Bayes Classifiers – Universitat de Barcelona and Institut d'Investigació en Intelligència Artificial
40. Hamine, V. and Helman, P. - A Theoretical and Experimental Evaluation of Augmented Bayesian – The University of New Mexico
41. Cerquides, J and Màntaras, R. L. - TAN Classifiers Based on Decomposable Distributions – Springer Science, Machine Learning, No. 59, 1-32 – 2005
42. Davis, J., Costa, V. S.,Ong, I. M., Page, D. and Dutra, I. - Using Bayesian Classifiers to Combine Rules – University of Madison- Wisconsin
43. Chow, C. K. and Liu, C. N. - Approximating discrete probability distributions with dependence trees – IEEE Transactions on Information Theory – Vol. IT 14, No. 3, May 1968
44. Sonka, M., Hlavac, V. and Boyle, R. – Image Processing, Analysis and Machine Vision – Brooks/Cole Publishing Company, USA – 1999
45. Parker, J. R. – Algorithms for Image Processing and Computer Vision – Wiley Computer Publishing – 1997.

46. Cowell, J. and Weston, J – Effects of ethnicity on skin detection – De Montfort University, 2005
47. Menzies et al. - The Performance of SolarScan - Arch Dermatol. November 2005;141:1388-1396
48. Yu, C. and Goshtasby, A. - A Picture Retrieval System Based on Contents - http://www.cs.pitt.edu/~panos/idm98/Imported/agoshtas.html
49. Walls, J., Tehrani, H., Cotton, S., Moncrieff, M., Hall, P.N. - The Non-Contact SIAscope in the Diagnosis of Cutaneous Lesions - American Association of Dermatology Meeting, March 2006
50. Tkalcic, M. and Tasic, I. - Colour Spaces – University of Ljubljana – 2002
51. Powell, M., Sarkar, M., Goldgof, D. and Ivanov, K. - A Methodology for Extracting Objective Color From Images – IEEE Transactions on Systems, Man and Cybernetics – Part B: Cybernetics, Volume 34,No. 5 – October 2004
52. Cohen, I. and Goldszmit, M. - Properties and benefits of calibrated classifiers – Hewlett-Packard Research Laboratories
53. Sharma, G. and Trussell, H. - Digital Color Imaging – IEEE Transactions on Image Processing, Vol. 6, No. 7 – 1997
54. Sigurdsson, S., Hansen, L. and Drzewiecki, K. - Color segmentation of skin lesions with the generalized Gaussian mixture model – National Hospital of Denmark and Technical University of Denmark
55. http://www.hcanc.org.br/dmeds/pele/pele1.html
56. http://www.cancer.org/docroot/STT/stt_0.asp
57. http://www.dfci.harvard.edu/can/screening
58. http://www.cancer.org/downloads/STT/CAFF2006PWSecured.pdf
59. Hsu, C., Chang, C. and Lin, C. - A Practical Guide to Support Vector Classification – National Taywan University
60. Burges, C. - A Tutorial on Support Vector Machines for Pattern Recognition – Kluwer Academic Publishers, Boston – 1998
61. Gunn, S. - Support Vector Machines for Classification and Regression – University of Southampton – 1998
62. Cover, T. and Hart, P. – Nearest Neighbor Pattern Classification – IEEE Transactions on Information Theory, Vol. IT-13, No. 1 – January 1967

63. Kulkarni, S., Lugosi, G and Venkatesh, S. – Learning Pattern Classification – A Survey – IEEE Transactions on Information Theory, Vol. 44, No. 6 – January 1998
64. Skowron, A. and Wojna, A. – K Nearest Neighbor Classification with Local Induction of the Simple Value Difference Metric – Warsaw University – 1997
65. Keller, J., Grey, M. and Givens, J. - A Fuzzy K-Nearest Neighbor Algorithm – IEEE Transactions on SMC, Volume SMC-15, No. 4 - 1985
66. Kerwin, M. - A Fuzzy K-Nearest Neighbor Algorithm: Review and Critical Analysis – November 2005
67. Bian, H and Mazlack, L. - Fuzzy-Rough Nearest-Neighbor Classification Approach – University of Cincinnati
68. Rabiner, L. - A tutorial on hidden Markov models and selected applications in Speech Recognition – Proceedings of the IEEE, Vol. 77, No. 2 – February 1989
69. Anderson, B and Moore, A. - Active Learning for Hidden Markov Models_Objective Functions and Algorithms – Proceedings of the 22nd International Conference on Machine Learning – 2005
70. Xu, M, Jackowsky, M., Goshtasby, A., Roseman, D., Bines, S., Yu, C., Dhawan, A. and Huntley, A. – Segmentation of skin cancer images – Image and Vision Computing – Elsevier – 1999
71. Cowell, J. & Weston, J. - Effects of Ethnicity on Skin Detection - Centre for Computational Intelligence - De Montfort University
72. Marr, D and Hildreth, E. - Theory of edge detection - Proc. Royal Soc. Lond., volume B 207, pages 187--217, 1980.
73. Canny, J. - A Computational Approach to Edge Detection - IEEE Trans. Pattern Analysis and Machine Intelligence, 8:679-714, 1986.
74. Rice, S. - Mathematical Analysis of Random Noise. *Bell System Technical Journal.* 23: 282-332. 24: 46-156. – 1944
75. Kheng, L.W. – Color Spaces and Color-Difference Equations – National University of Singapore, 2002
76. Stokes, M., Anderson, M., Chandrasekar, S. and Motta, R. - A Standard Default Color Space for the Internet – sRGB - http://www.w3.org/Graphics/Color/sRGB.html - 1996
77. Helterbrand, J. D., - One-Pixel-Wide Closed Boundary Identification – IEEE Transactions on Image Processing, Vol. 5, Nº 5, 1996

78. Goshtasby, A., - Geometric Modelling using rational Gaussian curves and surfaces – Computer Aided Design, Vol.27, Nº5, pp. 363-375 – Elsevier Science – 1995
79. Kononenko, I. and Bratko, I. - Information based evaluation criterion for classifier's performance, Machine Learning Journal, Vol.6, pp.67-80 – 1991
80. Cohen, J. - A coefficient of agreement for nominal scales - Education and Psychology Measures, Volume 20, Number 1, pp. 37-46. - 1960
81. Sim, J. and Wright, C. - The Kappa Statistic in Reliability Studies: Use, Interpretation, and Sample Size Requirements - Physical Therapy. Volume 85. Number 3, pp. 257-268 – 2005
82. Viera, A and Garrett, J. - Understanding Interobserver Agreement: The Kappa Statistic - Family Medicine Volume 37, Number 5, pp. 360-363 – 2005
83. Michie, D, Spiegelhalter, D. and Taylor, C. - Machine Learning, Neural and Statistical Classification – 1994
84. Braun, R., French, L. and Saurat, J.-H. - Dermoscopy of pigmented lesions: a valuable tool in the diagnosis of melanoma - Swiss Med Wkly Number 134, pp. 83–90 - 2004
85. Abbasi et al - Early Diagnosis of Cutaneous Melanoma: Revisiting the ABCD Criteria – The Journal of the American Medical Association - December 8, Volume 292, Number 22, pp. 2771-2776
86. Zhang, H., Fritts, J. & Goldman, S. - An Entropy-based Objective Evaluation Method for Image Segmentation - Dept. of Computer Science and Engineering, Washington University
87. Kröse, B. & van der Smagt, P. - An Introduction to Neural Networks - University of Amsterdam and German Aerospace Research Establishment - Eighth edition, November 1996
88. Rosenblatt, F. - The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain - Cornell Aeronautical Laboratory - Psychological Review, v65, No. 6, pp. 386-408.
89. Hopfield, J. - Neural networks and physical systems with emergent collective computational abilities - Proceedings of the National Academy of Sciences, Vol. 79, pp. 2554-2558, April 1982
90. Rojas, R. - Neural Networks - Springer-Verlag – Berlin – 1996
91. Sutton, R. & Barto, A. – Reinforcement Learning: An Introduction – MIT Press – 2005
92. Ghahramani, Z. – Unsupervised Learning - Gatsby Computational Neuroscience Unit, University College London, UK – 2004

93. Dayan, P. - Unsupervised Learning - The MIT Encyclopedia of the Cognitive Sciences – 1999
94. Ruzon, M. & Tomasi, C - Color Edge Detection with the Compass Operator - In IEEE Conference on Computer Vision and Pattern Recognition '99 - Volume 2, pp. 160-166 - June 1999.
95. http://www.dermatoscopes.com/ABCDE-melanoma-algorithm.shtml - ABCDE melanoma evaluation algorithm
96. Wong, W. & Chung, A. - Bayesian Image Segmentation Using Local Iso-intensity Structural Orientation - IEEE Transactions on image processing, Vol. 14, No. 10, October 2005
97. Young, I., Gerbrands, J. & Vliet, L. - Fundamentals of Image Processing - Delft University of Technology, Netherlands – 1998
98. Sharma, M. - Performance Evaluation of Image Segmentation and Texture Extraction Methods in Scene Analysis - University of Exeter – 2001
99. Su, Z. - Automatic Image Orientation Detection - The University Of Sheffield – 2004
100. Hamid, A., Allaoui, R & Sbihi, A. - A New Unsupervised Color Image Segmentation Algorithm - ICGST International Journal on Graphics, Vision and Image Processing (GVIP)
101. Wu, H-S, Deligdisch, L & Gil, J. - Segmentation of microscopic nuclear images – Recent Research Developments in Electronics - Transworld Research Network – 2004
102. Wang, S., Ge, F. & Liu, T. - Evaluating Edge Detection through Boundary Detection - Hindawi Publishing Corporation - EURASIP Journal on Applied Signal Processing - Volume 2006, Article ID 76278, Pages 1–15
103. Ben Hamza, A. & Krim, H. - Nonlinear Image Filtering: Trade-Off between Optimality and Practicality - North Carolina State University – 2001
104. Ben Hamza, A. & Krim, H. - Nonlinear Image Filtering in a Mixture of Gaussian and Heavy-Tailed Noise
105. Ben Hamza, A. & Krim, H. - Image Denoising: A Nonlinear Robust Statistical Approach - IEEE Transactions on Signal Processing, Vol. 49, No. 12, December 2001
106. Buckley, R. & Beretta, G. - Color Imaging on the Internet - University of Italian Switzerland (USI) – 2003
107. Yanai, K & Barnard, K. - Image Region Entropy: A Measure of "Visualness" of Web Images Associated with One Concept – Singapore, MM'05, November 6–11, 2005.

108. Brink, A. & Pendock, N., - Minimum cross-entropy threshold selection - Pattern Recognition 29, pp. 179–188 - 1996.
109. Pal, N. - On minimum cross-entropy thresholding - Pattern Recognition 29, pp. 575–580 - 1996.
110. Moser, G. & Serpico, S. - Generalized Minimum-Error Thresholding for Unsupervised Change Detection From SAR Amplitude Imagery - IEEE Transactions on Geoscience and Remote Sensing, Vol. 44, No. 10, October 2006.
111. Xue, J.-H. & Titterington, D. - Discriminative Image Thresholding - University of Glasgow – 2007.
112. Chidiac, H. & Ziou, D. - Classification of Image Edges - Vision Interface '99, Trois-Rivières, Canada.
113. Michie, D., Spiegelhalter, D. & Taylor, C. - Machine Learning, Neural and Statistical Classification – 1994
114. Hsu, C.-W., Chang, C.-C. & Lin, C.-J. - A Practical Guide to Support Vector Classification - National Taiwan University – 2008
115. School of Computer Science & Software Engineering - Edge Detection - The University of Western Australia
116. Nadernejad, E., Sharifzadeh, S. & Hassanpour, H. - Edge Detection Techniques: Evaluations and Comparisons - Applied Mathematical Sciences, Vol. 2, 2008, no. 31, 1507 – 1520
117. Nalwa, & Binford, T. - On Detecting Edges – Stanford University – 1986
118. Archambeau, C., Butz, T., Popovici, V., Verleysen, M. & Thiran, J.-P. - Supervised Nonparametric Information Theoretic Classification – IEEE 2004.
119. Rasche, K., Geist, R. & Westa, J. - Detail Preserving Reproduction of Color Images for Monochromats and Dichromats - IEEE Computer Society - May/June 2005
120. Kittler, H., Pehamberger, H., Wolff, K. & Binder, M. - Diagnostic accuracy of dermoscopy - THE LANCET Oncology Vol 3 March 2002
http://oncology.thelancet.com
121. Kittler, J. & Illingworth, J. - Minimum error thresholding - Patern Recognition, Vol. 19, No. 1, pp. 41—47 – 1986
122. http://www.libpng.org/pub/png/png-sitemap.html#info
123. http://www.libpng.org/pub/png/pngintro.html

124. Srinivasa, N., Ramakrishnan, K. & Rajgopal, K. - Detection of Edges from Projections - IEEE Transactions on Medical Imaging, Vol. 11, No. 1, March 1992
125. Escoda, O. & Vandergheynst, P. - Segmentation of Natural Images Using Scale-Space Representation with Multi-Scale Edge Supervised Hierarchical Linking - Swiss Federal Institute of Techonlogy in Lausanne (EPFL)
126. Tweddle, I. - The prickly genius- Colin MacLaurin (1698-1746) - The Mathematical Gazette Vol 82 No 495 - November 1998
127. Granville, W., Smith, P. & Longley, W. – Elements of the Differential and Integral Calculus - Ginn and Company – Boston
128. Hoffmann, G. - CIELab Color Space - The University of Applied Sciences - Oldenburg
129. Richter, K. - Linear Relationship between CIELAB and Device Coordinates for a new Colorimetric Image Technology (CIT) - BAM and TU Berlin Federal Institute for Materials Research and Testing (BAM) – 2005
130. Wnukowicz, K. & Skarbek, W. - Colour temperature estimation algorithm for digital images - properties and convergence - Opto-Electronics Review 11(3), 193-196 – 2003
131. Lilley, C., Lin, F., Hewitt, W. & Howard, T. - Colour in Computer Graphics - University of Manchester
132. Liao. W.-H. & Aggarwal, J. - Curve and Surface Interpolation Using Rational Radial Basis Functions - 13th International Conference on Pattern Recognition (ICPR'96) - Volume 4, p. 8 – 1996
133. Davis, T. - Homogeneous Coordinates and Computer Graphics - http://www.geometer.org/mathcircles - 2001
134. Reinhard, E. - High dynamic range imaging: Acquisition, Display, and Image-Based Lighting - Morgan Kaufmann – 2006
135. Kahn, E. & Reinhard, E. - A Survey of Color Spaces for Shadow Identification - University of Central Florida – 2004
136. Süsstrunka, S., Holmb, J. & Finlaysonc, G. - Chromatic Adaptation Performance of Different RGB Sensors - IS&T/SPIE Electronic Imaging - SPIE Vol. 4300 – 2001
137. Alessi, P., Fairchild, M., Hashimoto, K., Hunt, R., Luo, M., Mori, L., Nayatani, Y., Seim, T., Sobagaki, H. & Richter, K. - The CIE 1997 Interim Colour Appearance Model - CIE TC1-34 – 1998

138. Fairchild, M. - RLAB: a color appearance space for color reproduction - Proc. SPIE, Vol. 1909, 19 – 1993
139. Dash, M. & Liu, H. - Feature Selection for Classification – National University of Singapore – 1997
140. Amaro, J. – Cancro Cutâneo – Factos e Números – Associação Portuguesa de Cancro Cutâneo
141. Santos, F. – Cancro Cutâneo - Associação Portuguesa de Cancro Cutâneo
142. Cheng, Y., Swamisai, R., Umbaugh, S, Moss, R., Stoecker, W., Teegala, S. and Srinivasan, S. - Skin lesion classification using relative color features - Skin Research and Technology - Singapore – 2008
143. Hintz-Madsen, M., Hansen, L., Larsen, J., Olesen, E. and Drzewiecki, K. - Design and Evaluation of Neural Classifiers Application to Skin Lesion Classification - Proceedings of the 1995 IEEE Workshop on Neural Networks for Signal Processing V
144. She, Z. and Fish, P. - Skin Lesion Differentiation Using Skin Line Direction - School of Informatics, University of Wales
145. Sigurdsson, S., Larsen, J., Hansen, L., Philipsen, P and Wulf, H. - Outlier Estimation and Detection Application to Skin Lesion Classification - International conference on acoustics, speech and signal processing - Vol. 1 pp. 1049-1052 – 2002
146. Vincent, L and Soille, P. – Watersheds in Digital Spaces: An Efficient Algorithm Based on Immersion Simulations – IEEE Transactions om Pattern Analysis and Machine Intelligence, Vol 13, Nº 6, June 1991
147. Intel Image Processing Library Ref Manual – Intel Corporatiom – 1998
148. Beucher, S. and Lantuéjoul, C. - Use of watersheds in contour detection – International Workshop on Image Processing: Real Time Edge and Motion detection/estimation – 1979
149. Cocosco, C, Zijdenbos, A and Evans, A. - A Fully Automatic and Robust Brain MRI Tissue Classification Method - Medical Image Analysis, Vol.7 (4), p513-527 - Dec 2003.
150. Belongie, S., Malik, J. and Puzicha, J. - Matching Shapes - Eighth IEEE International Conference on Computer Vision - July 2001
151. Fortson, R., Lynch, D. and Newell, J. h Center - Automated Segmentation of Scleroderma in High Resolution CT Imagery - Los Alamos National Lab
152. Yang, A., Wright, J., Sastry S. and Ma, Y. - Unsupervised Segmentation of Natural Images via Lossy Data Compression – UC Berkeley – 2007

153. Cheriet, M., Kharma, N. Liu, C and Suen, C. - Character Recognition Systems: A Guide for Students and Practicioners – Wiley-Interscience – 2007
154. Celebi, M. et al. - A methodological approach to the classification of dermoscopy images - Computerized Medical Imaging and Graphics 31 (2007) 362–373 – Elsevier – 2007
155. Messadi, M, Bessaid, A. & Taleb-Ahmed, A. - Extraction of specific parameters for skin tumour classification - Journal of Medical Engineering & Technology, Vol. 33, No. 4, May 2009, 288–295
156. Marozas, M. & Jurkonis, R - Review on skin lesion imaging, analysis and automatic classification – Kaunas University of Technology - Biomedical Engineering 12th International Conference – Lithuania - 23-24 October 2008
157. Sigurdsson, S., Philipsen, P., Hansen, L., Larsen, J., Gniadecka, M. & Wulf, H. - Detection of skin cancer by classification of Raman spectra - IEEE Transactions on Biomedical Engineering Volume 51, Issue 10, Oct. 2004 Page(s):1784 - 1793

8 Appendixes

8.1 Appendix A - Data collected from the whole training set

		Red			Green			Blue			Hue			Saturation			Brightness		
Type	File name	Min	Avg	Max	Min	Avg	Max	Min	Avg	Max	Min	Avg	Max	Min	Avg	Max	Min	Avg	Max
AC	q0002a.png	163	196	207	111	139	153	84	132	141	0	6.56	358.8	0.23	0.35	0.44	0.49	0.64	0.68
AC	q0002a.png	83	155	180	69	109	128	42	83	100	3.24	21.67	350.4	0.13	0.3	0.43	0.25	0.47	0.54
AC	q0002a.png	135	144	250	95	107	206	68	98	190	2	11.74	30.57	0.15	0.19	0.88	0.42	0.47	0.85
AC	q0007a.png	94	104	235	59	73	179	31	42	155	1.02	30	359.06	0.21	0.42	0.69	0.27	0.29	0.76
AC	q0007a.png	187	225	241	116	174	188	93	147	163	0.74	20.77	26.53	0.37	0.57	0.75	0.55	0.73	0.78
AC	q0007a.png	86	229	250	44	171	209	29	134	179	0	23.37	356.33	0.25	0.65	0.9	0.23	0.71	0.84
AC	q0041a.png	148	166	217	95	96	164	41	62	114	13.2	19.62	35.35	0.34	0.46	0.61	0.39	0.45	0.65
AC	q0041a.png	77	159	246	35	94	179	22	64	160	0	18.95	359.06	0.23	0.43	0.83	0.22	0.44	0.8
AC	q0041a.png	148	192	197	95	125	140	41	73	107	13.2	26.22	37.86	0.34	0.49	0.59	0.39	0.52	0.59
AC	q0041a.png	147	231	252	85	165	202	34	131	170	9.33	20.4	32.5	0.35	0.68	0.94	0.35	0.71	0.82
AC	q0041a.png	93	131	191	54	88	124	18	43	96	3.1	30.68	35.33	0.29	0.51	0.7	0.23	0.34	0.55
AC	q0042a.png	94	115	138	74	87	113	32	40	75	20.36	37.6	63.27	0.21	0.48	0.5	0.25	0.3	0.41
AC	q0042a.png	85	147	159	61	103	112	25	58	86	5	30.34	48.39	0.22	0.43	0.6	0.23	0.4	0.48
AC	q0042a.png	82	145	191	57	83	144	29	62	116	6.77	15.18	42.86	0.16	0.4	0.55	0.25	0.41	0.6
CH	q0023a.png	169	224	239	114	145	160	68	114	122	11.71	16.91	36.98	0.31	0.64	0.79	0.47	0.66	0.7
CH	q0023a.png	161	196	255	86	134	223	62	95	199	6.51	23.17	30.28	0.35	0.46	1	0.44	0.57	0.89
CH	q0023a.png	132	214	222	93	151	156	63	110	113	11.71	23.65	33.45	0.26	0.56	0.63	0.38	0.64	0.65
CH	q0024a.png	160	215	232	132	158	182	97	131	149	7.87	19.29	51.11	0.18	0.51	0.68	0.51	0.68	0.73
CH	q0024a.png	149	197	252	102	141	212	86	126	191	3.48	12.68	31.9	0.27	0.38	0.92	0.46	0.63	0.86
CH	q0024a.png	106	165	181	68	122	143	49	88	110	0	26.49	35.84	0.22	0.3	0.43	0.3	0.5	0.57
CH	q0025a.png	108	142	231	79	111	180	53	80	161	0	30	358.8	0.15	0.28	0.62	0.32	0.44	0.77
CH	q0025a.png	198	219	239	147	176	200	107	144	176	10.91	25.6	33.63	0.42	0.51	0.7	0.6	0.71	0.81
CH	q0025a.png	153	186	204	112	131	152	78	100	130	6.56	21.63	37.63	0.19	0.38	0.49	0.47	0.56	0.64
CH	q0025a.png	108	156	240	79	99	198	53	88	169	0	9.71	358.8	0.15	0.28	0.73	0.32	0.48	0.79
CH	q0026a.png	165	185	216	103	129	149	84	114	142	0	12.68	359.19	0.26	0.34	0.54	0.49	0.59	0.68
CH	q0026a.png	172	190	216	109	128	157	84	115	137	0	10.4	359.17	0.31	0.37	0.55	0.53	0.6	0.69
CH	q0026a.png	172	204	225	109	137	162	73	128	139	2.31	7.11	25.63	0.34	0.43	0.63	0.48	0.65	0.7
CH	q0026a.png	154	172	195	91	109	130	59	78	107	2.54	19.79	31.84	0.28	0.38	0.47	0.44	0.49	0.59
IN	q0008a.png	168	253	255	74	171	194	65	147	152	8.5	13.58	355	0.37	0.96	1	0.48	0.78	0.8
IN	q0008a.png	233	255	255	158	212	212	111	180	185	15.15	25.6	31.96	0.73	1	1	0.67	0.85	0.86
IN	q0008a.png	15	167	255	13	101	187	10	79	151	2.54	15	358.57	0.1	0.36	1	0.06	0.48	0.8
IN	q0015a.png	224	244	255	148	181	189	112	140	162	10.63	23.65	28	0.62	0.83	1	0.66	0.75	0.82
IN	q0015a.png	148	214	236	92	150	172	63	102	144	11.37	25.71	32	0.3	0.58	0.72	0.43	0.62	0.75
IN	q0015a.png	138	182	217	88	122	152	59	96	122	10.71	18.14	33.55	0.29	0.37	0.58	0.39	0.55	0.66
IN	q0015a.png	221	254	255	145	187	193	112	145	162	8.76	23.12	26.84	0.57	0.98	1	0.66	0.78	0.82
IN	q0015a.png	140	191	255	87	127	193	55	100	162	4.44	17.8	28.8	0.27	0.42	1	0.38	0.57	0.81
IN	q0018a.png	233	254	255	142	174	175	109	139	145	8.33	18.26	24.38	0.72	0.98	1	0.67	0.77	0.78
IN	q0018a.png	131	152	231	74	88	148	46	61	116	1.67	17.8	28.13	0.28	0.43	0.71	0.35	0.42	0.68
IN	q0018a.png	158	234	255	101	146	183	81	122	155	8.28	12.86	21.72	0.32	0.73	1	0.47	0.7	0.8
IN	q0019a.png	166	225	225	101	157	159	61	118	128	12.39	21.87	28.98	0.37	0.64	0.64	0.45	0.67	0.67
IN	q0019a.png	111	161	205	87	111	144	56	86	120	6.1	20	38.44	0.18	0.3	0.48	0.35	0.48	0.63
IN	q0019a.png	145	220	236	93	153	175	67	124	150	2.81	18.13	29.27	0.26	0.58	0.73	0.42	0.67	0.75
IN	q0029a.png	202	244	255	136	168	211	77	118	157	20.5	23.81	36.48	0.53	0.85	1	0.56	0.71	0.81
IN	q0029a.png	143	222	222	82	140	151	48	90	106	16.25	22.73	32.86	0.38	0.67	0.67	0.37	0.61	0.64
IN	q0029a.png	196	252	255	124	185	224	86	133	188	14.4	26.22	32.24	0.47	0.95	1	0.56	0.75	0.87
IN	q0030a.png	162	223	247	102	144	181	50	85	123	16.89	25.65	32.5	0.39	0.68	0.89	0.42	0.6	0.73
IN	q0030a.png	191	204	255	112	132	192	69	84	142	13.85	24	30.68	0.48	0.54	1	0.51	0.56	0.78
IN	q0030a.png	102	173	184	58	112	124	19	68	75	11.54	25.14	33.16	0.38	0.44	0.69	0.24	0.47	0.51
IN	q0039a.png	178	192	252	116	148	191	73	101	153	12.13	30.99	32.62	0.33	0.42	0.94	0.5	0.57	0.79
IN	q0039a.png	164	215	217	108	157	158	55	111	121	15	26.54	33.16	0.37	0.57	0.58	0.44	0.64	0.66
IN	q0039a.png	159	189	227	108	138	164	69	95	127	15	27.45	37.24	0.29	0.42	0.67	0.45	0.56	0.69
IN	q0039a.png	163	255	255	112	206	210	73	173	177	11.71	24.15	30.99	0.3	1	1	0.49	0.84	0.85
OA	q0003a.png	231	255	255	147	216	217	68	148	159	22.8	38.13	38.13	0.76	1	1	0.59	0.79	0.81
OA	q0003a.png	142	192	242	54	127	162	38	73	97	3.19	27.23	35.45	0.4	0.49	0.85	0.35	0.52	0.66

		Red			Green			Blue			Hue			Saturation			Brightness		
Type	File name	Min	Avg	Max	Min	Avg	Max	Min	Avg	Max	Min	Avg	Max	Min	Avg	Max	Min	Avg	Max
OA	q0003a.png	140	201	255	67	130	192	49	68	121	17.48	27.97	33.57	0.44	0.55	1	0.38	0.53	0.74
OA	q0009a.png	207	251	255	117	173	202	82	134	179	7.11	20	25.63	0.57	0.94	1	0.57	0.75	0.85
OA	q0009a.png	225	243	255	133	181	202	111	142	177	7.11	23.17	25.63	0.65	0.81	1	0.67	0.75	0.85
OA	q0009a.png	126	143	195	75	89	129	54	65	107	0.97	18.46	30.75	0.22	0.38	0.45	0.36	0.41	0.58
OA	q0009a.png	122	217	234	64	135	157	43	121	134	0.67	8.75	356.31	0.29	0.56	0.71	0.34	0.66	0.72
OA	q0017a.png	221	240	251	131	158	175	104	134	151	4.75	13.58	23.59	0.61	0.76	0.93	0.64	0.73	0.79
OA	q0017a.png	176	205	245	103	135	166	84	110	149	1.91	15.79	25.14	0.32	0.49	0.83	0.53	0.62	0.77
OA	q0017a.png	210	236	255	118	159	198	103	129	167	4.83	16.82	23.12	0.54	0.74	1	0.62	0.72	0.87
OA	q0017a.png	131	214	233	76	135	157	56	120	135	0	9.57	357.72	0.23	0.53	0.71	0.37	0.65	0.71
OA	q0017a.png	120	179	255	49	113	192	50	91	177	0	15	359.19	0.27	0.37	1	0.34	0.53	0.85
OA	q0020a.png	185	233	255	114	172	207	65	128	177	11.25	25.14	31.06	0.45	0.7	1	0.49	0.71	0.85
OA	q0020a.png	114	166	209	75	117	139	44	84	105	4.62	24.15	38.82	0.21	0.33	0.57	0.31	0.49	0.61
OA	q0020a.png	162	196	218	100	130	143	65	95	115	9.33	20.79	29.22	0.35	0.46	0.61	0.45	0.57	0.65
OA	q0022a.png	119	145	183	66	96	112	4	29	53	19.09	34.66	38.18	0.48	0.67	0.94	0.26	0.34	0.45
OA	q0022a.png	124	214	227	54	132	142	0	72	79	17.94	25.35	32.11	0.51	0.63	1	0.25	0.56	0.59
OA	q0022a.png	99	165	168	32	91	109	0	26	49	10.91	28.06	36.92	0.5	0.73	1	0.21	0.37	0.42
OA	q0027a.png	203	240	254	143	181	206	84	139	170	18.13	24.95	33.67	0.51	0.77	0.98	0.57	0.74	0.83
OA	q0027a.png	175	232	236	123	162	171	80	128	135	15	19.62	33	0.36	0.69	0.74	0.51	0.71	0.72
OA	q0027a.png	171	248	255	106	186	208	67	147	181	8	23.17	27.08	0.38	0.88	1	0.47	0.77	0.85
OA	q0028a.png	217	255	255	129	168	180	77	116	131	17.65	22.45	29.78	0.64	1	1	0.58	0.73	0.76
OA	q0028a.png	200	234	255	120	158	209	60	106	169	17.21	24.36	33.71	0.56	0.75	1	0.51	0.67	0.83
OA	q0028a.png	183	239	255	115	156	182	52	116	127	16.75	19.51	31.84	0.47	0.79	1	0.46	0.7	0.75
OA	q0028a.png	161	249	255	97	176	220	56	123	182	13.59	25.24	33.79	0.45	0.91	1	0.47	0.73	0.86
OA	q0033a.png	216	229	247	141	156	176	119	141	163	3.57	10.23	21.43	0.54	0.63	0.86	0.66	0.73	0.79
OA	q0033a.png	84	201	223	100	134	157	72	107	129	8.73	17.23	134.79	0.21	0.47	0.62	0.43	0.6	0.68
OA	q0033a.png	178	216	237	119	152	169	100	124	153	0	18.26	359.25	0.31	0.54	0.71	0.55	0.67	0.76
OA	q0037a.png	190	212	238	116	136	173	65	86	113	16.95	23.81	32.95	0.48	0.59	0.79	0.51	0.58	0.69
OA	q0037a.png	117	174	191	69	106	121	24	58	78	13.1	25.86	36.12	0.36	0.5	0.67	0.28	0.45	0.53
OA	q0037a.png	143	147	224	69	76	150	19	36	101	13.33	22.7	32.07	0.39	0.61	0.77	0.33	0.36	0.64
OA	q0040a.png	229	240	253	168	182	199	133	144	176	10.34	23.75	29.01	0.63	0.76	0.96	0.71	0.75	0.83
OA	q0040a.png	206	226	252	133	164	193	110	143	178	5.07	15.16	26.47	0.48	0.59	0.93	0.62	0.72	0.84
OA	q0040a.png	166	235	236	114	173	177	84	160	160	1.67	10.4	357.5	0.26	0.65	0.7	0.5	0.77	0.77
WH	q0001a.png	227	254	255	151	175	207	129	158	196	1.67	10.63	23.45	0.59	0.98	1	0.7	0.81	0.88
WH	q0001a.png	145	213	220	93	136	152	70	119	142	0	12.13	357.5	0.22	0.53	0.58	0.44	0.65	0.7
WH	q0001a.png	167	183	252	102	122	192	87	104	182	0	13.67	359.19	0.28	0.35	0.92	0.5	0.56	0.85
WH	q0004a.png	229	247	255	178	196	206	144	165	187	10.29	24.15	30.99	0.62	0.84	1	0.73	0.81	0.86
WH	q0004a.png	233	250	255	178	202	206	146	180	186	10.29	18.86	30.99	0.64	0.88	1	0.75	0.84	0.85
WH	q0004a.png	178	211	220	136	157	162	98	123	136	13.01	23.18	32	0.32	0.5	0.58	0.55	0.65	0.69
WH	q0004a.png	189	241	255	118	173	208	88	150	188	5.25	15.16	25.63	0.42	0.76	1	0.54	0.77	0.87
WH	q0004a.png	155	226	230	113	158	163	84	135	141	2.54	15.16	31.27	0.23	0.61	0.65	0.49	0.71	0.72
WH	q0005a.png	203	225	237	149	170	188	121	150	168	2.81	16	28.42	0.41	0.56	0.69	0.64	0.74	0.79
WH	q0005a.png	139	186	202	94	119	141	75	110	131	0	7.11	359.19	0.19	0.36	0.43	0.44	0.58	0.65
WH	q0005a.png	157	193	220	111	136	165	70	117	150	3.81	15	29.14	0.25	0.38	0.53	0.45	0.61	0.72
WH	q0006a.png	211	233	255	126	168	231	97	138	217	10.32	18.95	32.96	0.52	0.68	1	0.6	0.73	0.92
WH	q0006a.png	142	172	184	101	118	132	65	80	106	9.23	24.78	34.5	0.25	0.37	0.43	0.42	0.49	0.56
WH	q0006a.png	112	190	219	51	130	153	33	94	124	1.5	22.5	359.19	0.29	0.42	0.59	0.29	0.56	0.66
WH	q0010a.png	215	237	252	160	177	214	132	149	193	3.75	19.09	29.19	0.49	0.71	0.92	0.69	0.76	0.86
WH	q0010a.png	141	171	198	114	123	146	72	101	124	8.4	18.86	45.82	0.17	0.29	0.41	0.44	0.53	0.63
WH	q0010a.png	117	172	252	70	116	213	46	99	194	0	13.97	358.46	0.15	0.31	0.92	0.34	0.53	0.87
WH	q0011a.png	203	232	249	113	148	158	88	112	130	6.41	18	22.24	0.49	0.72	0.91	0.58	0.67	0.74
WH	q0011a.png	203	233	255	113	128	158	88	109	130	3.66	9.19	21.43	0.49	0.74	1	0.58	0.67	0.74
WH	q0011a.png	184	254	255	98	146	171	78	118	145	5.28	12.35	355.25	0.39	0.99	1	0.53	0.73	0.78
WH	q0011a.png	252	255	255	158	211	218	125	180	190	12.43	24.8	29.29	0.95	1	1	0.74	0.85	0.87
WH	q0011a.png	193	243	255	111	138	161	74	116	144	6.32	10.39	22.24	0.44	0.84	1	0.54	0.7	0.78

		Red			Green			Blue			Hue			Saturation			Brightness		
Type	File name	Min	Avg	Max	Min	Avg	Max	Min	Avg	Max	Min	Avg	Max	Min	Avg	Max	Min	Avg	Max
WH	q0012a.png	243	255	255	146	192	209	111	174	195	9.43	13.33	25.5	0.84	1	1	0.69	0.84	0.86
WH	q0012a.png	147	220	238	85	126	153	63	114	132	1.18	6.79	26.51	0.29	0.6	0.76	0.42	0.65	0.73
WH	q0012a.png	167	205	222	99	127	140	82	107	129	0	12.24	357.75	0.33	0.49	0.58	0.49	0.61	0.69
WH	q0012a.png	175	197	246	86	116	161	72	97	139	2.05	11.4	21.43	0.38	0.46	0.87	0.48	0.58	0.74
WH	q0012a.png	238	255	255	141	190	208	110	155	195	6.14	21	24.71	0.79	1	1	0.68	0.8	0.86
WH	q0013a.png	220	250	255	118	154	223	73	112	204	11.89	18.26	27.46	0.66	0.93	1	0.58	0.71	0.9
WH	q0013a.png	97	141	195	44	56	112	30	51	91	0	3.33	359.12	0.31	0.47	0.62	0.26	0.38	0.56
WH	q0013a.png	196	209	255	100	129	182	67	92	153	4.17	18.97	24.89	0.51	0.56	1	0.52	0.59	0.8
WH	q0013a.png	197	221	255	100	117	190	67	90	150	5.13	12.37	23.92	0.5	0.66	1	0.52	0.61	0.79
WH	q0014a.png	198	245	255	139	191	210	105	167	194	6.56	18.46	28.04	0.42	0.8	1	0.59	0.91	0.98
WH	q0014a.png	144	160	224	114	115	163	90	96	147	5	17.81	34.44	0.12	0.25	0.57	0.47	0.5	0.73
WH	q0014a.png	137	191	242	80	144	196	69	126	177	0	16.62	356.36	0.25	0.34	0.75	0.4	0.62	0.92
WH	q0014a.png	119	152	187	81	115	141	65	106	126	1.54	11.74	358.57	0.17	0.18	0.34	0.37	0.51	0.61
WH	q0016a.png	182	220	240	116	150	169	66	114	134	15.26	20.39	28.42	0.45	0.6	0.78	0.49	0.65	0.73
WH	q0016a.png	131	196	206	84	118	138	46	98	104	12.14	12.24	36.15	0.29	0.45	0.55	0.36	0.58	0.59
WH	q0016a.png	172	224	255	91	152	199	68	114	158	6.12	20.73	27.78	0.36	0.64	1	0.47	0.66	0.81
WH	q0021a.png	195	242	249	124	184	186	99	172	183	0	10.29	359.25	0.38	0.73	0.87	0.6	0.91	0.93
WH	q0021a.png	148	206	250	85	124	180	67	113	172	0	7.1	359.25	0.28	0.49	0.89	0.42	0.63	0.93
WH	q0021a.png	168	226	255	85	151	182	67	128	175	0	14.09	359.39	0.36	0.63	1	0.46	0.69	0.84
WH	q0031a.png	185	230	255	103	150	207	91	123	194	0	15.14	359.35	0.37	0.68	1	0.56	0.69	0.88
WH	q0031a.png	166	217	217	99	142	147	73	123	128	3.57	12.13	32	0.3	0.55	0.56	0.47	0.67	0.67
WH	q0031a.png	164	235	255	82	138	218	61	131	195	0	4.04	359.3	0.3	0.72	1	0.44	0.72	0.88
WH	q0031a.png	92	131	185	47	77	116	38	65	99	0	10.91	359.17	0.22	0.34	0.54	0.27	0.38	0.55
WH	q0032a.png	199	210	240	114	143	166	103	126	152	1.24	12.14	356.7	0.41	0.48	0.75	0.6	0.66	0.76
WH	q0032a.png	131	206	206	89	151	151	65	121	121	8.11	21.18	33.53	0.23	0.46	0.47	0.39	0.64	0.64
WH	q0032a.png	122	155	174	75	114	126	69	96	116	1.18	18.31	358.93	0.17	0.24	0.35	0.37	0.49	0.55
WH	q0034a.png	199	224	234	140	172	179	100	132	166	12	26.09	34.11	0.43	0.6	0.69	0.59	0.7	0.77
WH	q0034a.png	199	207	229	140	150	178	100	121	146	15	20.23	34.11	0.43	0.47	0.64	0.59	0.64	0.73
WH	q0034a.png	164	223	233	126	167	190	101	144	157	9.38	17.47	39.05	0.2	0.55	0.66	0.53	0.72	0.75
WH	q0034a.png	153	185	215	109	144	163	89	122	153	5	20.95	38.77	0.22	0.31	0.48	0.48	0.6	0.72
WH	q0035a.png	205	252	255	119	194	205	83	172	195	2.98	16.5	23.18	0.51	0.93	1	0.57	0.93	0.96
WH	q0035a.png	149	183	221	104	117	136	69	93	125	2.93	16	30.34	0.25	0.38	0.62	0.43	0.54	0.66
WH	q0035a.png	150	188	255	104	137	191	84	106	166	5.65	22.68	27.33	0.27	0.38	1	0.46	0.58	0.93
WH	q0036a.png	222	234	255	145	166	194	121	143	177	4.34	15.16	23.45	0.54	0.68	1	0.68	0.74	0.95
WH	q0036a.png	205	217	227	149	164	183	102	120	139	16.77	27.22	36.4	0.47	0.56	0.64	0.61	0.66	0.71
WH	q0036a.png	215	227	254	143	167	196	115	139	177	2.89	19.09	27.86	0.54	0.61	0.98	0.65	0.72	0.94
WH	q0038a.png	211	231	241	148	186	195	116	165	184	1.79	19.09	28.48	0.46	0.58	0.73	0.64	0.78	0.93
WH	q0038a.png	150	200	225	107	123	156	83	113	142	0	6.9	358.42	0.25	0.44	0.6	0.46	0.61	0.71
WH	q0038a.png	188	243	250	123	185	204	106	165	192	0	15.39	359.17	0.34	0.76	0.89	0.58	0.8	0.96

8.2 Appendix B – Minima, Averages and Maxima

Type	FileName	AvgOfRed_min	AvgOfRed_avg	AvgOfRed_max	AvgOfGreen_min	AvgOfGreen_avg	AvgOfGreen_max	AvgOfBlue_min	AvgOfBlue_avg	AvgOfBlue_max
AC	q0002a.png	127	165	212.3333333	91.66666667	118.3333333	162.3333333	64.66666667	104.3333333	143.6666667
AC	q0007a.png	122.3333333	186	242	73	139.3333333	192	51	107.6666667	165.6666667
AC	q0041a.png	122.6	175.8	220.6	72.8	113.6	161.8	31.2	74.6	129.4
AC	q0042a.png	87	135.6666667	162.6666667	64	91	123	28.66666667	53.33333333	92.33333333
CH	q0023a.png	154	211.3333333	238.6666667	97.66666667	143.3333333	179.6666667	64.33333333	106.3333333	144.6666667
CH	q0024a.png	138.3333333	192.3333333	221.6666667	100.6666667	140.3333333	179	77.33333333	115	150
CH	q0025a.png	141.75	175.75	228.5	104.25	129.25	182.5	72.75	103	159
CH	q0026a.png	165.75	187.75	213	103	125.75	149.5	75	108.75	131.25
IN	q0008a.png	138.6666667	225	255	81.66666667	161.3333333	197.6666667	62	135.3333333	162.6666667
IN	q0015a.png	174.2	217	243.6	112	153.4	179.8	80.2	116.6	150.4
IN	q0018a.png	174	213.3333333	247	105.6666667	136	168.6666667	78.66666667	107.3333333	138.6666667
IN	q0019a.png	140.6666667	202	222	93.66666667	140.3333333	159.3333333	61.33333333	109.3333333	132.6666667
IN	q0029a.png	180.3333333	239.3333333	244	114	164.3333333	195.3333333	70.33333333	113.6666667	150.3333333
IN	q0030a.png	151.6666667	200	228.6666667	90.66666667	129.3333333	165.6666667	46	79	113.3333333
IN	q0039a.png	166	212.75	237.75	111	162.25	180.75	67.5	120	144.5
OA	q0003a.png	171	216	250.6666667	96	157.6666667	190.3333333	51.66666667	96.33333333	125.6666667
OA	q0009a.png	170	213.5	234.75	97.25	144.5	172.5	72.5	115.5	149.25
OA	q0017a.png	171.6	214.8	247.8	95.4	140	177.6	79.4	116.8	159.8
OA	q0020a.png	153.6666667	198.3333333	227.3333333	96.33333333	139.6666667	163	58	102.3333333	132.3333333
OA	q0022a.png	114	174.6666667	192.6666667	50.66666667	106.3333333	121	1.333333333	42.33333333	60.33333333
OA	q0027a.png	183	240	248.3333333	124	176.3333333	195	77	138	162
OA	q0028a.png	195.25	244.25	255	115.25	164.5	197.75	61.25	115.25	152.25
OA	q0033a.png	159.3333333	215.3333333	235.6666667	120	147.3333333	167.3333333	97	124	148.3333333
OA	q0037a.png	150	177.6666667	217.6666667	84.66666667	107.3333333	148	36	60	97.33333333
OA	q0040a.png	200.3333333	233.6666667	247	138.3333333	173	189.6666667	109	149	171.3333333
WH	q0001a.png	179.6666667	216.6666667	242.3333333	115.3333333	145	183.6666667	95.33333333	127	173.3333333
WH	q0004a.png	196.8	235	243	144.6	177.6	189	112	150.6	167.6
WH	q0005a.png	166.3333333	201.3333333	219.6666667	118	141.6666667	164.6666667	88.66666667	125.6666667	149.6666667
WH	q0006a.png	155	198.3333333	219.3333333	92.66666667	138.6666667	172	65	104	149
WH	q0010a.png	157.6666667	193.3333333	234	114.6666667	138.6666667	191	83.33333333	116.3333333	170.3333333
WH	q0011a.png	207	243.4	253.8	118.6	154.2	173.2	90.6	127	147.8
WH	q0012a.png	194	226.4	243.2	111.4	150.2	174.2	87.6	129.4	153.8
WH	q0013a.png	177.5	205.25	240	90.5	114	174.25	59.25	86.25	149.5
WH	q0014a.png	149.5	187	227	103.5	141.25	177.5	82.25	123.75	161
WH	q0016a.png	161.6666667	213.3333333	233.6666667	97	140	168.6666667	60	108.6666667	132
WH	q0021a.png	170.3333333	224.6666667	251.3333333	98	153	182.6666667	77.66666667	137.6666667	176.6666667
WH	q0031a.png	151.75	203.25	228	82.75	126.75	172	65.75	110.5	154
WH	q0032a.png	150.6666667	190.3333333	206.6666667	92.66666667	136	147.6666667	79	114.3333333	129.6666667
WH	q0034a.png	178.75	209.75	227.75	128.75	158.25	175	97.5	129.75	155.5
WH	q0035a.png	168	207.6666667	243.6666667	109	149.3333333	177.3333333	78.66666667	123.6666667	158.6666667
WH	q0036a.png	214	226	245.3333333	145.6666667	165.6666667	191	112.6666667	134	164.3333333
WH	q0038a.png	183	224.6666667	238.6666667	126	164.6666667	185	101.6666667	147.6666667	172.6666667

Type	FileName	AvgOfHue_min	AvgOfHue_max	AvgOfSat_min	AvgOfSat_avg	AvgOfSat_max	AvgOfBri_min	AvgOfBri_avg	AvgOfBri_max
AC	q0002a.png	1.746666667	246.59	0.17	0.28	0.583333333	0.386666667	0.526666667	0.69
AC	q0007a.png	0.586666667	247.3066667	0.276666667	0.546666667	0.78	0.35	0.576666667	0.793333333
AC	q0041a.png	7.766	100.02	0.31	0.514	0.734	0.316	0.492	0.682
AC	q0042a.png	10.71	51.50666667	0.196666667	0.436666667	0.55	0.243333333	0.37	0.496666667
CH	q0023a.png	9.976666667	33.57	0.306666667	0.553333333	0.806666667	0.43	0.623333333	0.746666667
CH	q0024a.png	3.783333333	39.61666667	0.223333333	0.396666667	0.676666667	0.423333333	0.603333333	0.72
CH	q0025a.png	4.3675	197.215	0.2275	0.3625	0.635	0.4275	0.5475	0.7525
CH	q0026a.png	1.2125	193.9575	0.2975	0.38	0.5475	0.485	0.5825	0.665
IN	q0008a.png	8.73	248.51	0.4	0.773333333	1	0.403333333	0.703333333	0.82
IN	q0015a.png	9.182	29.838	0.41	0.636	0.86	0.504	0.654	0.772
IN	q0018a.png	6.093333333	24.74333333	0.44	0.713333333	0.903333333	0.496666667	0.63	0.753333333
IN	q0019a.png	7.1	32.23	0.27	0.506666667	0.616666667	0.406666667	0.606666667	0.683333333
IN	q0029a.png	17.05	33.86	0.46	0.823333333	0.89	0.496666667	0.69	0.773333333
IN	q0030a.png	14.09333333	32.11333333	0.416666667	0.553333333	0.86	0.39	0.543333333	0.673333333
IN	q0039a.png	13.46	33.5025	0.3225	0.6025	0.7975	0.47	0.6525	0.7475
OA	q0003a.png	14.49	35.71666667	0.533333333	0.68	0.95	0.44	0.613333333	0.736666667
OA	q0009a.png	3.965	109.58	0.4325	0.6725	0.79	0.485	0.6425	0.75
OA	q0017a.png	2.298	157.752	0.394	0.582	0.894	0.5	0.65	0.798
OA	q0020a.png	8.4	33.03333333	0.336666667	0.496666667	0.726666667	0.416666667	0.59	0.703333333
OA	q0022a.png	15.98	35.73666667	0.496666667	0.676666667	0.98	0.24	0.423333333	0.486666667
OA	q0027a.png	13.71	31.31666667	0.416666667	0.78	0.906666667	0.516666667	0.74	0.8
OA	q0028a.png	16.8	32.28	0.53	0.8625	1	0.505	0.7075	0.8
OA	q0033a.png	4.1	171.8233333	0.353333333	0.546666667	0.73	0.546666667	0.666666667	0.743333333
OA	q0037a.png	15.12666667	33.71333333	0.41	0.566666667	0.743333333	0.373333333	0.463333333	0.62
OA	q0040a.png	5.693333333	137.66	0.456666667	0.666666667	0.863333333	0.61	0.746666667	0.813333333
WH	q0001a.png	0.556666667	246.7133333	0.363333333	0.62	0.833333333	0.546666667	0.673333333	0.81
WH	q0004a.png	8.276	30.176	0.446	0.718	0.846	0.612	0.756	0.798
WH	q0005a.png	2.206666667	138.9166667	0.283333333	0.433333333	0.55	0.51	0.643333333	0.72
WH	q0006a.png	7.016666667	142.2166667	0.353333333	0.49	0.673333333	0.436666667	0.593333333	0.713333333
WH	q0010a.png	4.05	144.49	0.27	0.436666667	0.75	0.49	0.606666667	0.786666667
WH	q0011a.png	6.82	90.09	0.552	0.858	0.982	0.594	0.724	0.782
WH	q0012a.png	3.76	91.18	0.526	0.71	0.842	0.552	0.696	0.776
WH	q0013a.png	5.2975	108.8475	0.495	0.655	0.905	0.47	0.5725	0.7625
WH	q0014a.png	3.275	194.3525	0.24	0.3925	0.665	0.4575	0.61	0.76
WH	q0016a.png	11.17333333	30.78333333	0.366666667	0.563333333	0.776666667	0.44	0.63	0.71
WH	q0021a.png	0	359.2966667	0.34	0.616666667	0.92	0.493333333	0.71	0.833333333
WH	q0031a.png	0.8925	277.455	0.2975	0.5725	0.775	0.435	0.615	0.745
WH	q0032a.png	3.51	249.72	0.27	0.393333333	0.523333333	0.453333333	0.596666667	0.65
WH	q0034a.png	10.345	36.51	0.32	0.4825	0.6175	0.5475	0.665	0.7425
WH	q0035a.png	3.82	26.95	0.343333333	0.563333333	0.873333333	0.486666667	0.65	0.783333333
WH	q0036a.png	8	29.23666667	0.516666667	0.616666667	0.873333333	0.646666667	0.706666667	0.8
WH	q0038a.png	0.596666667	248.69	0.35	0.593333333	0.74	0.56	0.73	0.8

Type	FileName	Red_min_avg	Red_avg_avg	Red_max_avg	Green_min_avg	Green_avg_avg	Green_max_avg	Blue_min_avg	Blue_avg_avg	Blue_max_avg
AC	q0002a.png	0.43	0.45	0.44	0.36	0.31	0.31	0.22	0.24	0.25
AC	q0002a.png	0.45	0.41	0.39	0.32	0.31	0.32	0.23	0.28	0.29
AC	q0002a.png	0.46	0.42	0.41	0.31	0.30	0.31	0.23	0.28	0.28
AC	q0007a.png	0.54	0.43	0.39	0.28	0.32	0.33	0.18	0.25	0.28
AC	q0007a.png	0.51	0.47	0.41	0.32	0.33	0.31	0.17	0.19	0.27
AC	q0007a.png	0.47	0.41	0.41	0.29	0.32	0.32	0.23	0.27	0.28
AC	q0041a.png	0.56	0.50	0.46	0.33	0.34	0.30	0.11	0.16	0.23
AC	q0041a.png	0.55	0.44	0.40	0.32	0.31	0.32	0.13	0.25	0.27
AC	q0041a.png	0.52	0.49	0.44	0.33	0.32	0.32	0.14	0.19	0.24
AC	q0041a.png	0.57	0.50	0.42	0.26	0.30	0.31	0.16	0.20	0.27
AC	q0041a.png	0.52	0.51	0.44	0.33	0.30	0.33	0.14	0.19	0.23
AC	q0042a.png	0.50	0.48	0.45	0.36	0.33	0.31	0.15	0.19	0.24
AC	q0042a.png	0.47	0.48	0.42	0.37	0.36	0.35	0.16	0.17	0.23
AC	q0042a.png	0.49	0.50	0.42	0.34	0.29	0.32	0.17	0.21	0.26
CH	q0023a.png	0.48	0.46	0.46	0.32	0.30	0.31	0.19	0.24	0.23
CH	q0023a.png	0.52	0.46	0.38	0.28	0.32	0.33	0.20	0.22	0.29
CH	q0023a.png	0.46	0.45	0.45	0.32	0.32	0.32	0.22	0.23	0.23
CH	q0024a.png	0.44	0.42	0.38	0.30	0.30	0.32	0.26	0.27	0.29
CH	q0024a.png	0.41	0.43	0.41	0.34	0.31	0.32	0.25	0.26	0.26
CH	q0024a.png	0.48	0.44	0.42	0.30	0.33	0.33	0.22	0.23	0.25
CH	q0025a.png	0.45	0.45	0.42	0.33	0.31	0.31	0.23	0.24	0.27
CH	q0025a.png	0.45	0.45	0.40	0.33	0.29	0.33	0.22	0.26	0.28
CH	q0025a.png	0.45	0.43	0.40	0.33	0.33	0.31	0.22	0.24	0.28
CH	q0025a.png	0.44	0.41	0.39	0.33	0.33	0.33	0.24	0.27	0.29
CH	q0026a.png	0.47	0.43	0.43	0.29	0.30	0.29	0.24	0.27	0.28
CH	q0026a.png	0.47	0.44	0.42	0.30	0.30	0.31	0.23	0.27	0.27
CH	q0026a.png	0.51	0.48	0.45	0.30	0.30	0.30	0.19	0.22	0.25
CH	q0026a.png	0.49	0.43	0.43	0.31	0.29	0.31	0.21	0.27	0.26
IN	q0008a.png	0.55	0.44	0.42	0.24	0.30	0.32	0.21	0.26	0.25
IN	q0008a.png	0.46	0.39	0.39	0.31	0.33	0.33	0.22	0.28	0.28
IN	q0008a.png	0.39	0.48	0.43	0.34	0.29	0.32	0.26	0.23	0.25
IN	q0015a.png	0.46	0.43	0.42	0.31	0.32	0.31	0.23	0.25	0.27
IN	q0015a.png	0.50	0.46	0.42	0.31	0.30	0.32	0.20	0.24	0.27
IN	q0015a.png	0.46	0.43	0.42	0.30	0.32	0.32	0.23	0.25	0.27
IN	q0015a.png	0.48	0.46	0.44	0.31	0.31	0.31	0.21	0.24	0.25
IN	q0015a.png	0.49	0.46	0.43	0.30	0.32	0.31	0.21	0.22	0.26
IN	q0018a.png	0.48	0.45	0.44	0.29	0.31	0.30	0.23	0.25	0.25
IN	q0018a.png	0.52	0.50	0.47	0.29	0.29	0.30	0.18	0.20	0.23
IN	q0018a.png	0.46	0.47	0.43	0.30	0.29	0.31	0.24	0.24	0.26
IN	q0019a.png	0.48	0.44	0.42	0.30	0.31	0.31	0.22	0.25	0.27
IN	q0019a.png	0.51	0.45	0.44	0.31	0.31	0.31	0.19	0.24	0.25
IN	q0019a.png	0.44	0.45	0.44	0.34	0.31	0.31	0.22	0.24	0.26
IN	q0029a.png	0.49	0.46	0.41	0.33	0.32	0.34	0.19	0.22	0.25
IN	q0029a.png	0.48	0.44	0.38	0.31	0.32	0.34	0.21	0.23	0.28
IN	q0029a.png	0.52	0.49	0.46	0.30	0.31	0.32	0.18	0.20	0.22
IN	q0030a.png	0.51	0.49	0.43	0.30	0.31	0.33	0.19	0.20	0.24
IN	q0030a.png	0.52	0.49	0.45	0.32	0.32	0.33	0.16	0.19	0.22
IN	q0030a.png	0.57	0.49	0.48	0.32	0.32	0.32	0.11	0.19	0.20
IN	q0039a.png	0.49	0.44	0.42	0.32	0.34	0.32	0.20	0.23	0.26
IN	q0039a.png	0.47	0.45	0.44	0.32	0.33	0.32	0.21	0.23	0.25
IN	q0039a.png	0.50	0.45	0.44	0.33	0.33	0.32	0.17	0.23	0.24
IN	q0039a.png	0.47	0.40	0.40	0.32	0.32	0.33	0.21	0.27	0.28
OA	q0003a.png	0.52	0.41	0.40	0.33	0.35	0.34	0.15	0.24	0.25
OA	q0003a.png	0.51	0.50	0.45	0.32	0.33	0.34	0.18	0.17	0.21
OA	q0003a.png	0.61	0.49	0.48	0.23	0.32	0.32	0.16	0.19	0.19
OA	q0009a.png	0.48	0.43	0.40	0.28	0.32	0.32	0.24	0.25	0.28
OA	q0009a.png	0.49	0.48	0.45	0.29	0.30	0.30	0.21	0.22	0.25
OA	q0009a.png	0.53	0.46	0.45	0.28	0.29	0.30	0.19	0.26	0.26
OA	q0009a.png	0.51	0.45	0.40	0.29	0.31	0.32	0.20	0.24	0.28
OA	q0017a.png	0.48	0.46	0.44	0.28	0.30	0.30	0.23	0.24	0.27
OA	q0017a.png	0.50	0.46	0.44	0.29	0.29	0.30	0.21	0.26	0.26
OA	q0017a.png	0.55	0.47	0.41	0.22	0.30	0.31	0.23	0.24	0.28
OA	q0017a.png	0.48	0.45	0.44	0.29	0.30	0.30	0.23	0.25	0.26
OA	q0017a.png	0.49	0.45	0.40	0.27	0.30	0.31	0.24	0.25	0.29
OA	q0020a.png	0.49	0.45	0.46	0.32	0.32	0.31	0.19	0.23	0.23
OA	q0020a.png	0.50	0.47	0.46	0.31	0.31	0.30	0.20	0.23	0.24
OA	q0020a.png	0.51	0.44	0.40	0.31	0.32	0.32	0.18	0.24	0.28
OA	q0022a.png	0.76	0.59	0.52	0.24	0.32	0.33	0.00	0.09	0.15
OA	q0022a.png	0.70	0.51	0.51	0.30	0.32	0.32	0.00	0.17	0.18
OA	q0022a.png	0.63	0.54	0.53	0.35	0.36	0.32	0.02	0.11	0.15
OA	q0027a.png	0.47	0.43	0.40	0.33	0.32	0.33	0.20	0.25	0.27
OA	q0027a.png	0.46	0.44	0.44	0.33	0.31	0.32	0.21	0.25	0.25
OA	q0027a.png	0.50	0.43	0.40	0.31	0.32	0.32	0.19	0.25	0.28
OA	q0028a.png	0.53	0.47	0.40	0.32	0.32	0.33	0.16	0.21	0.27
OA	q0028a.png	0.52	0.47	0.45	0.33	0.31	0.32	0.15	0.23	0.23
OA	q0028a.png	0.54	0.45	0.39	0.29	0.32	0.33	0.17	0.22	0.28
OA	q0028a.png	0.51	0.47	0.45	0.30	0.31	0.32	0.18	0.22	0.23
OA	q0033a.png	0.45	0.44	0.42	0.30	0.30	0.30	0.25	0.27	0.28
OA	q0033a.png	0.33	0.45	0.44	0.39	0.30	0.31	0.28	0.24	0.25
OA	q0033a.png	0.45	0.44	0.42	0.30	0.31	0.30	0.25	0.25	0.27
OA	q0037a.png	0.51	0.49	0.45	0.31	0.31	0.33	0.18	0.20	0.22
OA	q0037a.png	0.56	0.51	0.49	0.33	0.32	0.31	0.11	0.17	0.20
OA	q0037a.png	0.62	0.56	0.47	0.30	0.30	0.32	0.08	0.14	0.21
OA	q0040a.png	0.43	0.42	0.40	0.32	0.32	0.32	0.25	0.25	0.28
OA	q0040a.png	0.46	0.42	0.40	0.30	0.31	0.31	0.24	0.27	0.29
OA	q0040a.png	0.46	0.41	0.41	0.31	0.30	0.31	0.23	0.28	0.28

FileName	Red_min_avg	Red_avg_avg	Red_max_avg	Green_min_avg	Green_avg_avg	Green_max_avg	Blue_min_avg	Blue_avg_avg	Blue_max_avg
q0001a.png	0.45	0.43	0.39	0.30	0.30	0.31	0.25	0.27	0.30
q0001a.png	0.47	0.45	0.43	0.30	0.29	0.30	0.23	0.25	0.28
q0001a.png	0.47	0.45	0.40	0.29	0.30	0.31	0.24	0.25	0.29
q0004a.png	0.42	0.40	0.39	0.32	0.32	0.32	0.26	0.27	0.29
q0004a.png	0.42	0.40	0.39	0.32	0.32	0.32	0.26	0.28	0.29
q0004a.png	0.43	0.43	0.42	0.33	0.32	0.31	0.24	0.25	0.26
q0004a.png	0.48	0.43	0.39	0.30	0.31	0.32	0.22	0.27	0.29
q0004a.png	0.44	0.44	0.43	0.32	0.30	0.31	0.24	0.26	0.26
q0005a.png	0.43	0.41	0.40	0.32	0.31	0.32	0.26	0.28	0.28
q0005a.png	0.45	0.45	0.43	0.31	0.29	0.30	0.24	0.27	0.28
q0005a.png	0.46	0.43	0.41	0.33	0.30	0.31	0.21	0.26	0.28
q0006a.png	0.46	0.46	0.44	0.33	0.32	0.31	0.21	0.22	0.25
q0006a.png	0.57	0.46	0.44	0.26	0.31	0.31	0.17	0.23	0.25
q0006a.png	0.49	0.43	0.36	0.29	0.31	0.33	0.22	0.26	0.31
q0010a.png	0.42	0.42	0.38	0.32	0.31	0.32	0.26	0.26	0.29
q0010a.png	0.43	0.43	0.42	0.35	0.31	0.31	0.22	0.26	0.26
q0010a.png	0.50	0.44	0.38	0.30	0.30	0.32	0.20	0.26	0.29
q0011a.png	0.47	0.39	0.38	0.30	0.33	0.33	0.23	0.28	0.29
q0011a.png	0.50	0.47	0.46	0.28	0.30	0.29	0.22	0.23	0.24
q0011a.png	0.51	0.49	0.46	0.29	0.28	0.29	0.20	0.23	0.26
q0011a.png	0.51	0.49	0.45	0.27	0.28	0.30	0.22	0.23	0.25
q0011a.png	0.50	0.50	0.47	0.28	0.27	0.29	0.22	0.23	0.24
q0012a.png	0.49	0.43	0.39	0.29	0.32	0.32	0.22	0.26	0.29
q0012a.png	0.50	0.48	0.46	0.29	0.27	0.29	0.21	0.25	0.25
q0012a.png	0.48	0.47	0.45	0.28	0.29	0.29	0.24	0.24	0.26
q0012a.png	0.53	0.48	0.45	0.26	0.28	0.30	0.22	0.24	0.25
q0012a.png	0.49	0.41	0.39	0.29	0.31	0.32	0.22	0.28	0.29
q0013a.png	0.54	0.48	0.37	0.29	0.30	0.33	0.18	0.22	0.30
q0013a.png	0.54	0.52	0.44	0.27	0.27	0.31	0.18	0.21	0.26
q0013a.png	0.54	0.49	0.43	0.28	0.30	0.31	0.18	0.21	0.26
q0013a.png	0.57	0.57	0.49	0.26	0.23	0.28	0.18	0.21	0.23
q0014a.png	0.41	0.43	0.42	0.33	0.31	0.31	0.26	0.26	0.28
q0014a.png	0.48	0.41	0.39	0.28	0.31	0.32	0.24	0.27	0.29
q0014a.png	0.45	0.41	0.41	0.31	0.31	0.31	0.25	0.28	0.28
q0014a.png	0.45	0.41	0.39	0.31	0.32	0.32	0.24	0.28	0.29
q0016a.png	0.50	0.45	0.44	0.32	0.31	0.31	0.18	0.24	0.25
q0016a.png	0.50	0.48	0.46	0.32	0.29	0.31	0.18	0.24	0.23
q0016a.png	0.52	0.46	0.42	0.27	0.31	0.33	0.21	0.23	0.26
q0021a.png	0.47	0.40	0.40	0.30	0.31	0.30	0.24	0.29	0.30
q0021a.png	0.49	0.47	0.42	0.28	0.28	0.30	0.22	0.26	0.29
q0021a.png	0.53	0.45	0.42	0.27	0.30	0.30	0.21	0.25	0.29
q0031a.png	0.49	0.45	0.44	0.29	0.29	0.30	0.22	0.26	0.26
q0031a.png	0.53	0.47	0.38	0.27	0.27	0.33	0.20	0.26	0.29
q0031a.png	0.52	0.48	0.46	0.27	0.28	0.29	0.21	0.24	0.25
q0031a.png	0.49	0.46	0.39	0.27	0.30	0.32	0.24	0.24	0.30
q0032a.png	0.46	0.43	0.43	0.31	0.32	0.32	0.23	0.25	0.25
q0032a.png	0.46	0.42	0.42	0.28	0.31	0.30	0.26	0.26	0.28
q0032a.png	0.48	0.44	0.43	0.27	0.30	0.30	0.25	0.26	0.27
q0034a.png	0.42	0.42	0.41	0.32	0.31	0.32	0.26	0.27	0.28
q0034a.png	0.44	0.41	0.40	0.31	0.32	0.31	0.25	0.27	0.29
q0034a.png	0.45	0.42	0.40	0.32	0.33	0.31	0.23	0.25	0.29
q0034a.png	0.45	0.43	0.41	0.32	0.31	0.32	0.23	0.25	0.26
q0035a.png	0.50	0.41	0.40	0.29	0.31	0.32	0.20	0.28	0.29
q0035a.png	0.46	0.47	0.46	0.32	0.30	0.28	0.21	0.24	0.26
q0035a.png	0.44	0.44	0.42	0.31	0.32	0.31	0.25	0.25	0.27
q0036a.png	0.45	0.43	0.41	0.30	0.31	0.31	0.25	0.26	0.28
q0036a.png	0.45	0.43	0.41	0.33	0.33	0.33	0.22	0.24	0.25
q0036a.png	0.45	0.43	0.41	0.30	0.31	0.31	0.24	0.26	0.28
q0038a.png	0.44	0.40	0.39	0.31	0.32	0.31	0.24	0.28	0.30
q0038a.png	0.44	0.46	0.43	0.31	0.28	0.30	0.24	0.26	0.27
q0038a.png	0.45	0.41	0.39	0.29	0.31	0.32	0.25	0.28	0.30

8.3 Appendix C – Classifiers

8.3.1 Bayesian classifiers

As stated by Theodoridis and Koutrombas, [3] "given a classification task of **M** classes, C_1, C_2, …, C_m, and an unknown pattern, which is represented by a feature vector **x**, we form the **M** conditional probabilities $\mathbf{P}(C_i \mid \mathbf{x})$, i = 1, 2, … , **M**. Sometimes these are also referred to as posterior probabilities. In words, each of them represents the probability that the unknown pattern belongs to the respective class C_i, given that the corresponding feature vector takes the value **x**".

In fact a classification task consists of nothing more than trying to group a set of objects into a class C, according to some basic characteristics that are somewhat common to all of them – the feature vector. Since we can determine the values of every feature, according to Bayes rule:

$$P(C_i \mid x) = \frac{p(x \mid C_i)P(C_i)}{p(x)} \tag{8-1}$$

which can also be written as:

$$\text{posterior} = \frac{\text{likelihood} \times \text{prior}}{\text{normalizing constant}} \tag{8-2}$$

where **p(x)** is the probability distribution function of **x** and is given by:

$$p(x) = \sum_{i=1}^{n} p(x \mid C_i)P(C_i) \tag{8-3}$$

If we now consider $\boldsymbol{n} = 2$, the classification rule can be described as:

If $P(C_1 \mid x) > P(C_2 \mid x)$, **x** is classified as belonging to C_1

If $P(C_1 \mid x) < P(C_2 \mid x)$, **x** is classified as belonging to C_2

According to Bayes rule, we may also write the above expressions, $P(C_1 \mid x)$ and $P(C_2 \mid x)$ as:

$$p(x \mid C_1)P(C_1) \tag{8-4}$$

$$p(x \mid C_2)P(C_2) \tag{8-5}$$

Since we are dealing with probabilities, the best we can do is trying to maximize these values, in order to minimize the risk of getting a wrong classification. This issue can be better understood by the analysis of the following figure which was adapted from Theodoridis and Koutroumbas [3]. It represents two equiprobable classes and the variations of $p(x \mid C_i)$ for i = 1, 2 as functions of x for the case of a single feature. The line at point x0 represents the threshold partitioning the feature space in two regions.

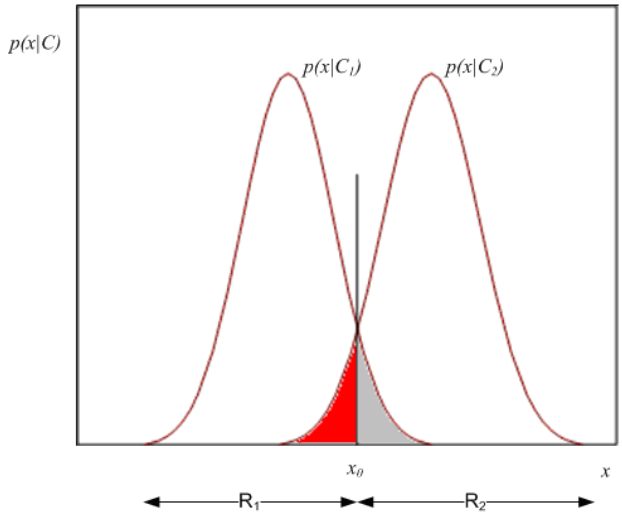


Figure 8-1 - Regions R1 and R2 formed by the Bayesian Classifier for two equiprobable classes

From figure 7-1 we can see that, all values of x belonging to R_1 will be classified as belonging to C_1. Every value of x in R_2 will be assigned to C_2. Obviously this will lead to errors. If x values fall either on the red or grey areas, they can be classified as belonging to class C_1 although lying on R_2 region, or vice-versa. A classification error will then occur if $x \in R_1$ although it belongs to C_2 or if $x \in R_2$ although it belongs to C_1.

In fact, the total probability of a wrong classification is given by:

$$2P_e = \int_{-\infty}^{x_0} p(x \mid C_2)dx + \int_{x_0}^{+\infty} p(x \mid C_1)dx \tag{8-6}$$

This corresponds to both the red and grey areas in 6-1.

One of this work's most important tasks must then be the minimization of this value in order to guaranty the most accurate classification possible. Although [3] the Bayesian classifier is optimal with respect to minimizing the classification error probability, this is not always the best criterion to be adopted for minimization. Depending on the relevant features, some classification errors are more important than others. To cope with this, it is sometimes better to weigh each error with a different penalty value. Assuming we have R_j regions - with the values of j between 1 and M – to be respectively assigned to classes C_j, and assuming that a feature vector x belonging to class C_k lies in R_i – with – $i \neq k$ – then, this vector will erroneously be classified as belonging to class C_i. If we then assign a penalty of λ_{ki} – known as loss – to every classification error between both k and i classes then we can create a loss matrix – L – with every penalty term. The loss or risk associated with C_k is then defined by:

$$r_k = \sum_{i=1}^{M} \lambda_{ki} \int_{R_i} p(x \mid C_k)dx \tag{8-7}$$

and the average risk of misclassification, given by:

$$\begin{aligned} r_k &= \sum_{k=1}^{M} r_k P(C_k) \quad \text{or} \\ r_k &= \sum_{i=1}^{M} \int_{R_i} \left(\sum_{k=1}^{M} \lambda_{ki}\, p(x \mid C_k) P(C_k) \right) dx \end{aligned} \tag{8-8}$$

must be minimized [3,28]. We can achieve this by minimizing each of the integrals. This is equivalent to selecting regions so that:

$$x \in R_i \quad \text{if} \quad l_i \equiv \sum_{k=1}^{M} \lambda_{ki}\, p(x \mid C_k) P(C_k) < l_j \equiv \sum_{k=1}^{M} \lambda_{kj}\, p(x \mid C_k) P(C_k) \qquad \forall j \neq i \tag{8-9}$$

If $\lambda_{ki} = 1 - \delta_{ki}$, with δ_{ki} being Kronecker's delta (0 if k ≠ i and 1 if k = i), then, the minimization of the classification error probability is achieved by minimizing the average loss.

When we consider only two classes, we may write:

$$l_1 = \lambda_{11} p(x \mid C_1) P(C_1) + \lambda_{21} p(x \mid C_2) P(C_2) \tag{8-10}$$

$$l_2 = \lambda_{12} p(x \mid C_1) P(C_1) + \lambda_{22} p(x \mid C_2) P(C_2) \tag{8-11}$$

and we assign x to C_1 if $l_1 < l_2$ which is the same as saying that:

$$(\lambda_{21} - \lambda_{22}) p(x \mid C_2) P(C_2) < (\lambda_{12} - \lambda_{11}) p(x \mid C_1) P(C_1) \tag{8-12}$$

If, as seems obvious, we penalize wrong decisions more that correct ones, then, $\lambda_{ij} > \lambda_{ii}$ and the rule stated before becomes:

$$x \in C_1 (C_2) \quad \text{if} \quad l_{12} \equiv \frac{p(x \mid C_1)}{p(x \mid C_2)} > (<) \frac{P(C_2)}{P(C_1)} \frac{\lambda_{21} - \lambda_{22}}{\lambda_{12} - \lambda_{11}} \tag{8-13}$$

The ratio l_{12} is known as the likelihood ratio and, assuming a loss matrix like:

$$L = \begin{bmatrix} 0 & \lambda_{12} \\ \lambda_{21} & 0 \end{bmatrix}$$

in which errors on classification of patterns that belong to class C2 are very problematic, we will have to choose values for λs such that $\lambda_{21} > \lambda_{12}$. This leads to assigning to class C2, patterns that respect the relation:

$$p(x \mid C_2) > p(x \mid C_1)\frac{\lambda_{12}}{\lambda_{21}} \qquad (8\text{-}14)$$

having been assumed that $P(C_1) = P(C_2) = 1/2$. This results in a translation of the threshold of figure 7-1 to the left of x_0, increasing, in fact, the size of region R_2.

From the various classifiers described within literature, by their simplicity and accuracy [19,40,42] we considered using the following:

- Naïve Bayes (NB);
- Tree Augmented Naïve Bayes (TAN).

The results achieved will be compared to the results of the following very largely studied algorithms:

- k Nearest Neighbour (kNN);
- Support Vector Machines (SVM);
- Neural Network.

8.3.1.1 **Naïve Bayes**

This is the most simple and most widely used Bayesian classifier and is based on the naïve assumptions that all the attributes used for classification are statistically independent. It is often used (as a simplifying assumption) in cases where the "effect variables are **not** conditionally independent given the cause variable" [26,27,37]. According to Russel and Norvig, in practice, Naïve Bayes systems can work surprisingly well when the independence assumption is not true. Rish and Domingos and Pazzani [27,29,30] prove that "although independence is generally a poor assumption, in practice, naïve Bayes often competes well with more sophisticated classifiers". With his work, Rish demonstrates that "low-entropy feature distributions yield good performance of naïve Bayes". He also shows that "*naïve Bayes works well for nearly-functional feature dependencies, thus reaching its best performance in two opposite cases: completely independent features (as expected) and functionally dependent features (which is surprising)*". Still according to Rish, "the accuracy of naïve Bayes is not directly correlated with the degree of feature dependencies measured as the class-conditional mutual information between the features. Instead, a better predictor of naïve Bayes accuracy is the amount of

information about the class that is lost because of the independence assumption"
The full joint distribution, according to this model, can be written as:

$$\mathrm{P}(Cause, Effect_1 ... Effect_n) = \mathrm{P}(Cause)\prod_i \mathrm{P}(Effect_i \mid Cause) \tag{8-15}$$

In a more detailed view [33], assume that you have X_1 to X_n attributes with discrete values used to predict a discrete class C. Given an example – training set – with observed attribute values x_1 to x_n, the optimal classification prediction is class value c, such that $P(C = c \mid X_1 = x_1 \wedge ... \wedge X_n = x_n)$ is maximal. By Bayes' rule, this probability is given by:

$$\frac{P(X_1 = x_1 \wedge ... \wedge X_n = x_n \mid C = c)P(C = c)}{P(X_1 = x_1 \wedge ... \wedge X_n = x_n)} \tag{8-16}$$

The value of P(C = c) can easily be estimated from the training data. $P(X_1 = x_1 \wedge ... \wedge X_n = x_n)$ is a constant for each class value c. This problem resides therefore on estimating $P(X_1 = x_1 \wedge ... \wedge X_n = x_n \mid C = c)$ from the training set data. Still according to the chain rule, this can be written as:

$$P(X_1 = x_1 \mid X_2 = x_2 \wedge ... \wedge X_n = x_n,\ C = c)P(X_2 = x_2 \wedge ... \wedge X_n = x_n \mid C = c) \tag{8-17}$$

Using the same rule, the second term can be written as:

$$P(X_2 = x_2 \mid X_3 = x_3 \wedge ... \wedge X_n = x_n,\ C = c)P(X_3 = x_3 \wedge ... \wedge X_n = x_n \mid C = c) \tag{8-18}$$

and so on, until we reach the n order factor.

If we assume that the outcome of every X_i is independent of the outcomes of every other X_j, given C the values above respect the following conditions:

$$\begin{aligned} &P(X_1 = x_1 \mid X_2 = x_2 \wedge ... \wedge X_n = x_n,\ C = c) = P(X_1 = x_1 \mid C = c) \\ &P(X_2 = x_2 \mid X_3 = x_3 \wedge ... \wedge X_n = x_n,\ C = c) = P(X_2 = x_2 \mid C = c) \\ &\cdot\ \cdot\ \cdot \\ &P(X_{n-1} = x_{n-1} \mid X_n = x_n,\ C = c) = P(X_{n-1} = x_{n-1} \mid C = c) \end{aligned} \tag{8-19}$$

then the above equation $P(X_1 = x_1 \wedge ... \wedge X_n = x_n \mid C = c)$ can be seen written as:

$$P(X_1 = x_1 \mid C = c) P(X_2 = x_2 \mid C = c) \cdot \cdot \cdot P(X_n = x_n \mid C = c) \tag{8-20}$$

Having reached this stage, every factor can now be estimated, from the training data, according to the following expression:

$$\hat{P}(X_i = x_i \mid C = c) = \frac{count(A_i = a_i \wedge C = c)}{count(C = c)} \tag{8-21}$$

It can be shown that this equation results in estimates that maximize the likelihood, i. e. the parameter probabilities values that maximize the probability of the training samples.

This approach has proven rather effective in several applications, and particularly in medical diagnostics. In fact, if the most likely class is chosen, the model's prediction can be seen as deterministic [26].

8.3.1.2 **Tree Augmented Naïve Bayes (TAN)**

As stated by Friedman, Geiger, and Goldszmidt [19], this method "*approximates the interactions between attributes by using a tree structure imposed on the naïve Bayes structure*". In their *Bayesian Network Classifiers* paper, the authors show that "*this approximation is optimal*" and can be "*learned in polynomial time*".

A Bayesian network – **B** – *"is an annotated directed acyclic graph that encodes a joint probability distribution over a set of random variables, U". It can be described by a pair* $B = \langle G, \Theta \rangle$ *with the first component, G, being a directed acyclic graph whose vertices correspond to the random variables*

X_1, … ,X_n, and whose edges represent direct dependencies between the variables. The graph G encodes independence assumptions: each variable Xi is of its non-descendants given its parents in G. The second component of the pair, namely Θ, represents the set of parameters that quantifies the network. It contains a parameter $\theta_{X_i|\prod_{X_i}} = P_B\left(X_i \mid \prod_{X_i}\right)$ *for each possible value* x_i *of* X_i*, and* $\prod_{x_i}$ *of* $\prod_{X_i}$ *denotes the set of parents of* X_i *in G. A Bayesian network –* ***B*** *- defines a unique joint probability distribution over U given by"*

$$P_B(X_1, \cdots, X_n) = \prod_{i=1}^{n} P_B\left(X_i \mid \prod_{X_i}\right) = \prod_{i=1}^{n} \theta_{X_i|\prod_{X_i}} \tag{8-22}$$

With naïve Bayes, we get a network like the one in figure 7-2, below. According to Davis et al. [42], in this case, we "*expect every learned clause to be related to a clause in the "true" theory. Hence, we would also expect that the way each learned clause classifies an example is somehow dependent on the example's true classification. This suggests a simple approach where we represent the outcome for each clause as a random variable, whose value depends on the example's classification*".

Since naïve Bayes assumes total independence between clauses and, since we, normally, "*expect clauses to be strongly related*", the Tree Augmented Naïve Bayes (TAN) approach – of which, figure 6-3 is an example tree - becomes rather an interesting one.

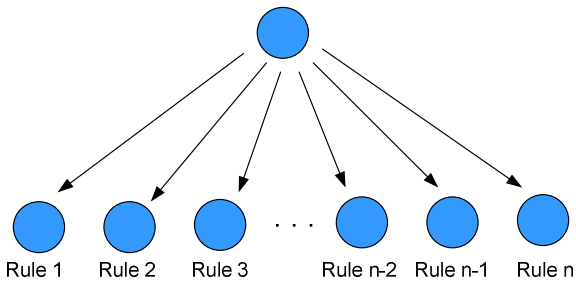


Figure 8-2 - Naïve Bayes Network

The TAN model [19,38,39][40,41,42], "*while retaining the basic structure of naïve Bayes, also permits each attribute to have at most, one other parent, allowing the model to capture dependencies between attributes. To decide which arcs to include in the "augmented" network, the algorithm makes*

a complete graph between all the non-class attributes, where the weight of each edge is given as the conditional mutual information between those two attributes. A maximum weight spanning tree is constructed over this graph and the edges that appear in the spanning tree are added to the network".

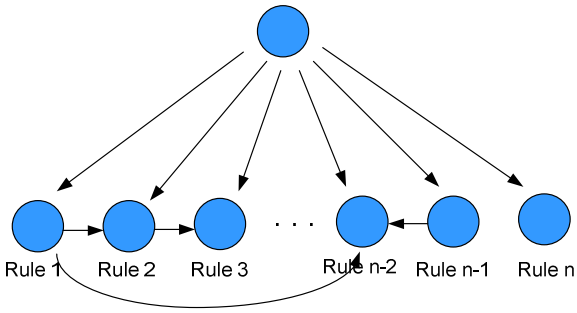


Figure 8-3 - Tree Augmented Bayes Network

A directed acyclic graph [19] on $\{X_1, \ldots, X_n\}$ is a tree if $\prod_{X_i}$ contains exactly one parent for all X_i, except for the root variable which has no parents. A function $\pi : \{1,\ldots, n\} \mapsto \{0,\ldots, n\}$ is said to define a tree over $X_1, \ldots, X_n$, if there is exactly one I such that $\pi(i) = 0$ - for the root of the tree - and there is no sequence $i_1, \ldots, i_k$ such that $\pi(i_j) = i_{j+1}$ for $i \le j \le k$ and $\pi(i_k) = i_1$, (i.e. with no cycles). Such a function defines a tree network where $\prod_{X_i} = \{X_{\pi(i)}\}$ if $\pi(i) = 0$.

Based on Chow and Liu work [43], we can describe an algorithm to construct the tree, with four steps:

Compute $I_{\hat{P}_D}(X_i; X_j)$ between each pair of variables, $i \neq j$, where:

$$I_p(\mathbf{X};\mathbf{Y}) = \sum_{x,y} P(x,y) \log \frac{P(x,y)}{P(x)P(y)} \tag{8-23}$$

is the *mutual information function* (this function measures how much information **Y** provides about **X**).

Build a complete undirected graph in which the vertices are the variables in **X**. Annotate the weight of an edge connecting X_i to X_j by $I_{\hat{P}_D}(X_i; X_j)$.

Build a maximum weighted spanning tree.

Transform the resulting undirected tree to a directed one by choosing a root variable and setting the direction of all edges to be outward from it.

Chow and Liu [43] proved that this algorithm finds the tree that maximizes the likelihood given the data ***D***.

If instead of using the original algorithm, we use *conditional mutual information* [19] between attributes given the class variable, we may define the function by:

$$I_P(\mathbf{X};\mathbf{Y}\mid\mathbf{Z})=\sum_{\mathbf{x},\mathbf{y},\mathbf{z}} P(\mathbf{x},\mathbf{y},\mathbf{z})\log\frac{P(\mathbf{x},\mathbf{y}\mid\mathbf{z})}{P(\mathbf{x}\mid\mathbf{z})P(\mathbf{y}\mid\mathbf{z})} \tag{8-24}$$

This function will measure the information that **Y** provides about **X** when the value of **Z** is known.

The above algorithm will now have the following form:

Compute $I_{\hat{P}_D}=(X_i;X_j\mid C)$ between each pair of attributes, $i\neq j$.

Build a complete tree in which the vertices are the attributes $X_1, \ldots, X_n$. Annotate the weight of an edge connecting Xi to Xj by $I_{\hat{P}_D}(X_i;X_j\mid C)$.

Build a maximum weighted spanning tree.

Transform the resulting undirected tree to a directed one by choosing a root variable and setting the direction of all edges to be outward from it.

Construct the TAN model by adding a vertex labelled by C and adding an arc from C to each X_i.

Assuming the TAN B_T is built, the value for the log likelihood will be given by:

$$LL(B_T\mid D)=N\cdot\sum_{X_i} I_{\hat{P}_D}\left(X_i;\prod X_i\right)\ +\ \text{constant term.} \tag{8-25}$$

Maximizing the log likelihood is thus, equivalent to maximizing:

$$\sum_{X_i} I_{\hat{P}_D}\left(X_i;\prod X_i\right) \tag{8-26}$$

Assuming a TAN B_T defined by $\pi(\cdot)$, since C has no parents, we get $I_{\hat{P}_D}\left(C;\prod_C\right)=0.$ Since the parents of X_i are defined by π, we set $I_{\hat{P}_D}\left(X_i;\prod X_i\right)=I_{\hat{P}_D}\left(X_i;X_{\pi(i)},C\right)$ if $\pi(i) > 0$ and $I_{\hat{P}_D}\left(X_i;\prod X_i\right)=I_{\hat{P}_D}\left(X_i;C\right)$ if $\pi(i) = 0$. We need then to maximize:

$$\sum_{i,\pi(i)>0} I_{\hat{P}_D}\left(X_i;X_{\pi(i)},C\right)+\sum_{i,\pi(i)=0} I_{\hat{P}_D}\left(X_i;C\right) \tag{8-27}$$

With some help from the chain rule, this term can be written as:

$$\sum_i I_{\hat{P}_D}\left(X_i;C\right)+\sum_{i,\pi(i)>0} I_{\hat{P}_D}\left(X_i;X_{\pi(i)}\mid C\right) \tag{8-28}$$

From this result we can see that the whole expression will be maximized if we maximize the second term. This is always accomplished by the TAN building algorithm that was described before.

8.3.2 k Nearest Neighbour (kNN)

Nearest Neighbour [26,62,63] models assume that the properties of any point *x* are likely to be similar to those of the points in the neighbourhood of *x*.

The algorithm for the Nearest Neighbour rule can be expressed like this [3]: "*Given an unknown feature vector* ***x*** *and a distance measure, then*

Out of the N Training vectors, identify the k nearest neighbours, irrespective of class label, k is chosen to be odd for a two class problem and in general, not to be multiple of the number of classes M.
Out of these k samples, identify the number of vectors, k_i, that belong to class C_i, i=1, 2, …,M.

Obviously, $\sum_{i=1}^{M} k_i = k$.

Assign x to the class Ci with the maximum number k_i of samples".

8.3.3 Fuzzy k-Nearest Neighbour (Fuzzy kNN)

Fuzzy set theory [65,66] "*replaces the* crisp is a member/is not a member *classification by assigning each sample a value of* ***how closely it represents each given class***. Making the techniques *fuzzy*, will then mean that *x* would have "*almost equal membership"* in class C_1 and in class C_2.

According to Bian and Mazlack [67], the Fuzzy k-Nearest Neighbour algorithm is composed by the following steps:

Getting the k nearest neighbours of the test pattern x

Let $X = \{ x_1, x_2, \ldots, x_n\}$ *be the set of training data and* $C = \{c_1, c_2, \ldots, cc\}$ *the result classification space. Let x be the data to be classified.*

Input x;

Set K, $1 \leq K \leq n$*;*

Set counter = 1;

For all $x_j \in X (1 \leq j \leq n)$ *Do*

Compute $\| x - x_j \|$

If ($i \leq K$)

Include x_j *in the set of K-NN and add 1 to counter*

Else if (x_j *is closer to x than any previous nearest neighbour)*

Begin

Delete the farthest of the K-NN

Include x_j *in the set of K-NN*

End

End For

Approximate x by the k-nearest neighbours

For all $c_j \in C (1 \leq i \leq c)$ *Do*

Compute $u_i(x)$

End For

Where

$$u_i(x) = \frac{\sum_{j=1}^{K} u_{i,j} \left\| x - x_j \right\|^{\frac{-2}{m-1}}}{\sum_{j=1}^{K} \left\| x - x_j \right\|^{\frac{-2}{m-1}}} \qquad (8\text{-}29)$$

In this expression, $u_{i,j}$ represents the membership of x_j to the ith class and m the level of weight to apply to the distance when each neighbour membership contribution is calculated.

8.3.4 Support Vector Machines (SVM)

8.3.4.1 Separable classes

Assuming a two class linearly separable situation [5,59,60], let x_i – with $i = 1, 2, ..., N$ – be the feature vectors of the training set, X. These points belong to either of classes C_1 or C_2. The objective will be the design of a hyper plane given by $g(x) = c^T x + c_0 = 0$ that classifies the training vectors, correctly.

If we define "*margins*" as the spaces left by the hyper plane between both classes then, from figure 8-4, we may determine that the margin for direction 1 is $2z_1$ and for direction 2, is $2z_2$ .The task will be to determine the direction that results in the maximum possible margin.

Since the distance of a point from a hyper plane is given by $z = \frac{|g(x)|}{\|\mathbf{c}\|}$, we can now scale c by c_0, so that the value of $g(x)$, at the nearest points in C_1 and C_2 (with circles in figure 8-4) is equal to +1 for C_1 and to -1 for C_2. This results in a margin of $\frac{2}{\|\mathbf{c}\|}$ and requiring that the following two conditions are true:

$$c^T x + c_0 \geq +1 \qquad \forall x \in C_1 \qquad (8\text{-}30)$$

$$c^T x + c_0 \leq -1 \qquad \forall x \in C_2 \tag{8-31}$$

Now, for each x_i, the corresponding class indicator will be denoted by y_i (+1 for class C_1 and -1 for class C_2) and we will have to calculate the parameters $\mathbf{c}$ and c_0 in such a way that the following function is minimized:

$$J(\mathbf{c}) \equiv \frac{1}{2}\| \mathbf{c} \|^2 \tag{8-32}$$

subject to $y_i\left(c^T x_i + c_0\right) \geq 1$, $i = 1, 2, \ldots, N$.

If we minimize $\| \mathbf{c} \|$, we obviously maximize the margin. This is an optimization task subject to Karush-Kuhn-Tucker [3] conditions, which are:

$$\frac{\partial}{\partial \mathbf{c}} \mathcal{L}(\mathbf{c}, c_0, \boldsymbol{\alpha}) = 0 \tag{8-33}$$

$$\frac{\partial}{\partial c_0} \mathcal{L}(\mathbf{c}, c_0, \boldsymbol{\alpha}) = 0 \tag{8-34}$$

$$\alpha_i \geq 0, \qquad i = 1, 2, \ldots, N \tag{8-35}$$

Where $\boldsymbol{\alpha}$ is the vector of the Lagrange multipliers, α_i and $\mathcal{L}(\mathbf{c}, c_0, \boldsymbol{\alpha})$ is a Lagrangian function defined by:

$$\mathcal{L}(\mathbf{c}, c_0, \boldsymbol{\alpha}) = \frac{1}{2}\mathbf{c}^T\mathbf{c} - \sum_{i=1}^{N} \alpha_i \left[y_i\left(\mathbf{c}^T x_i + c_0\right) - 1 \right] \tag{8-36}$$

Which, combined with $\frac{\partial}{\partial \mathbf{c}} \mathcal{L}(\mathbf{c}, c_0, \boldsymbol{\alpha}) = 0$, gives:

$$\mathbf{c} = \sum_{i=1}^{N} \alpha_i y_i x_i \qquad (8\text{-}37) \qquad \text{and} \qquad \sum_{i=1}^{N} \alpha_i y_i = 0 \qquad (8\text{-}38)$$

.

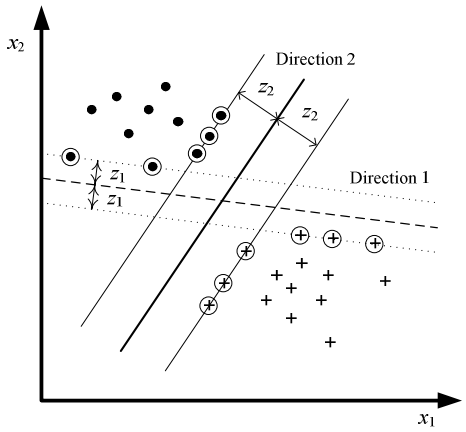


Figure 8-4 - Support Vectors (Separable Classes)

Since the Lagrange multipliers can be either zero or positive [3], the vector parameter **c** of the optimal solution is a linear combination of Ns ≤ N feature vectors associated with $\alpha_i \neq 0$. These are known as *support vectors* and the optimum hyperplane classifier as a *support vector machine*.

We now need to compute the involved parameters. This computation has been subject to a lot of studies that resulted in various algorithms [61].

From the above equations and since we want to minimize $J(\mathbf{c}) \equiv \frac{1}{2}\| \mathbf{c} \|^2$, we may reach an equivalent optimization task [3] that consists on achieving:

$$\max_{\alpha}\left(\sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j}^{N} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j \right) \qquad (8\text{-}39)$$

subject to $\sum_{i=1}^{N} \alpha_i y_i = 0$ and $\alpha \geq 0$ (8-49)

Once the optimal Lagrange multipliers are calculated, the optimal hyperplane is then obtained through the application of the conditions at the top of this page.

It should be taken into consideration that, although the resulting hyperplane is unique, there is no guarantee about the uniqueness of the Lagrangian multipliers.

8.3.4.2 **Nonseparable classes**

In this case the classes have quite diffuse borders and the above results may no longer be valid. Now we will have to consider three types of vectors:

Vectors falling within the exact area of a given class and correctly classified. To this kind of vectors, the above conditions can still be correctly applied;

Vectors falling inside the band between the support vectors and still correctly classified (see figure 8-5 below – points within squares). They satisfy the relation:

$$0 \leq y_i \left(\mathbf{c}^T x + c_0 \right) < 1 \tag{8-50}$$

Vectors that are misclassified (points within circles) and correspond to the inequality:

$$y_i \left(\mathbf{c}^T x + c_0 \right) < 0 \tag{8-51}$$

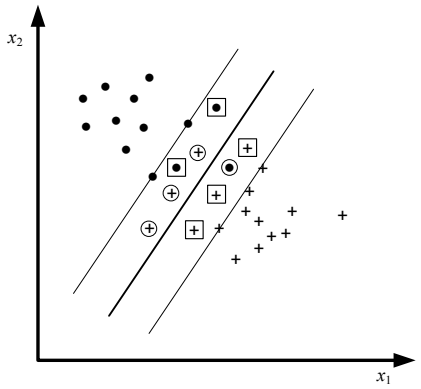


Figure 8-5 - Support Vectors (Nonseparable Classes)

If we introduce a new set of variables, all the vectors can be processed under a single type of constraints:

$$y_i\left[\mathbf{c}^T x + c_0\right] \geq 1 - \xi_i \tag{8-52}$$

Like this, the first kind of vectors correspond to $\xi_i = 0$, the second to $0 < \xi_i \leq 1$ and the last one to $\xi_i > 1$, These variables - ξ - are normally called *slack* variables.

Similarly to what we have done for the separable case, the objective here will be to choose a margin as large as possible but keeping the number of points with $\xi > 0$ as small as possible. This is equivalent to minimizing the function:

$$J(\mathbf{c}, c_0, \boldsymbol{\xi}) = \frac{1}{2}\| \mathbf{c} \|^2 + K\sum_{i=1}^{N} I(\xi_i) \tag{8-53}$$

Where K is a positive constant that determines the relative influence of the two terms, $\boldsymbol{\xi}$ is the vector of parameters ξ_i and:

$$I(\xi_i) = \begin{cases} 1 & \xi_i > 0 \\ 0 & \xi_i = 0 \end{cases} \tag{8-54}$$

Since $I(\xi_i)$ is a discontinuous function, we choose to work on a very similar function and the goal becomes:

Minimize: $$J(\mathbf{c}, c_0, \boldsymbol{\xi}) = \frac{1}{2}\| \mathbf{c} \|^2 + K\sum_{i=1}^{N} \xi_i \tag{8-55}$$

subject to: $$y_i\left[\mathbf{c}^T \mathbf{x}_i + c_0\right] \geq 1 - \xi_i, \qquad i = 1, 2, \ldots, N \tag{8-56}$$

and $$\xi_i \geq 0 \qquad i = 1, 2, \ldots, N \tag{8-57}$$

The corresponding Lagrangian is then:

$$\mathcal{L}(\mathbf{c}, c_0, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\mu}) = \frac{1}{2}\| \mathbf{c} \|^2 + K\sum_{i=1}^{N}\xi_i - \sum_{i=1}^{N}\mu_i\xi_i - \sum_{i=1}^{N}\alpha_i\left[y_i\left(\mathbf{c}^T\mathbf{x}_i + c_0\right) - 1 + \xi_i\right] \tag{8-58}$$

The corresponding KKT conditions become:

$$\frac{\partial \mathcal{L}}{\partial c} = 0 \qquad \text{or} \quad \mathbf{c} = \sum_{i=1}^{N}\alpha_i y_i \mathbf{x}_i \tag{8-59}$$

$$\frac{\partial \mathcal{L}}{\partial c_0} = 0 \qquad \text{or} \quad \sum_{i=1}^{N}\alpha_i y_i \tag{8-60}$$

$$\frac{\partial \mathcal{L}}{\partial \xi_i} = 0 \qquad \text{or} \quad K - \mu_i - \alpha_i = 0\,, \qquad i = 1, 2, \ldots, N \tag{8-61}$$

$$\alpha_i\left[y_i\left(\mathbf{c}^T\mathbf{x}_i + c_0\right) - 1 + \xi_i\right] = 0\,, \qquad i = 1, 2, \ldots, N \tag{8-62}$$

$$\mu_i\xi_i = 0\,, \qquad i = 1, 2, \ldots, N \tag{8-63}$$

$$\mu_i \geq 0\,, \qquad \alpha_i \geq 0\,, \qquad i = 1, 2, \ldots, N. \tag{8-64}$$

And the associated dual representation is:

$$\max_{\alpha}\left(\sum_{i=1}^{N}\alpha_i - \frac{1}{2}\sum_{i,j}^{N}\alpha_i\alpha_j y_i y_j \mathbf{x}_i^T\mathbf{x}_j\right) \tag{8-65}$$

subject to: $0 \leq \alpha_i \leq K\,, \qquad i = 1, 2, \ldots, N$ (8-66)

and to: $\sum_{i=1}^{N}\alpha_i y_i = 0\,.$ (8-67)

8.3.5 Neural Networks

Haykin wrote [5] that a "*neural network (NN), is a massive parallel distributed processor made up of simple processing units, which has a natural propensity for storing experiential knowledge and making it available for use. It resembles the brain in two respects:*

1. *Knowledge is acquired by the network from its environment through a learning process.*
2. *Interneuron connection strengths, known as synaptic weights, are used to store the acquired knowledge*".

According to Kröse and van der Smagt [87] "*an artificial network consists of a pool of simple processing units which communicate by sending signals to each other over a large number of weighted connections.*

A set of major aspects of a parallel distributed model can be distinguished:

- *A set of processing units ('neurons,' 'cells');*
- *A state of activation y_k for every unit, which is equivalent to the output of the unit;*
- *Connections between the units. Generally each connection is defined by a weight w_{jk} which determines the effect which the signal of unit j has on unit k;*
- *A propagation rule, which determines the effective input s_k of a unit from its external inputs;*
- *An activation function F_k, which determines the new level of activation based on the effective input sk(t) and the current activation $y_k(t)$ (i.e., the update);*
- *An external input (aka bias, offset) θ_k for each unit;*
- *A method for information gathering (the learning rule);*
- *An environment within which the system must operate, providing input signals and - if necessary - error signals*".

Haykin [5] defines a Neural Network as a **signal-flow directed graph** describing it as a network of directed links (**branches**) that are interconnected at certain points called **nodes**. A typical node **j** has an associated **node signal $\mathbf{x_j}$**. A typical directed link originates at node **j** and terminates on node **k**; it has an associated *transfer function* or *transmittance* that specifies the manner in which the node *k* depends on the signal x_j at node *j*. The flow of signals in the various parts of the graph is dictated by three basic rules:

1. A signal flows along a link only in the direction defined by the arrow on the link.
   Two different types of links may be distinguished:
   - **Synaptic links**, whose behaviour is governed by a linear input-output relation. Specifically, the node signal *x*, is multiplied by the synaptic weight w_{kj} to produce the node signal y_k, as seen below.

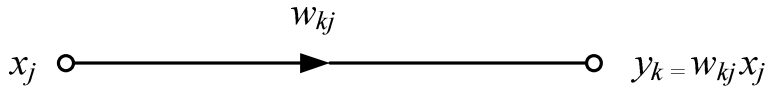


Figure 8-6 - Synaptic link

- **Activation links**, whose behaviour is governed in general by a nonlinear input-output relation. This form of relationship is best seen on the following figure, where $\varphi(\cdot)$ is the nonlinear activation function.

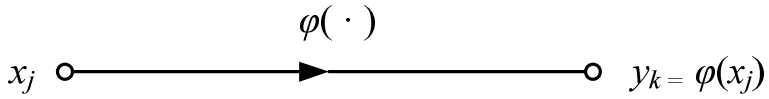


Figure 8-7 - Activation link

2. A node signal equals the algebraic sum of all signals entering the pertinent node via the incoming links.

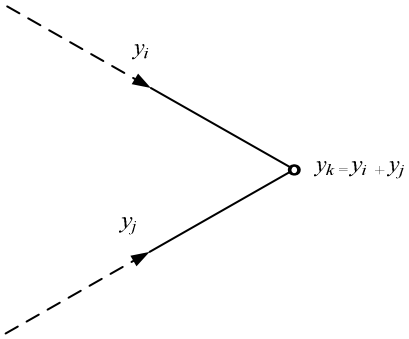


Figure 8-8 - Synaptic convergence or fan-in

3. The signal at a node is transmitted to each outgoing link originating from that node, with the transmission being entirely independent of the transfer functions of the outgoing links.

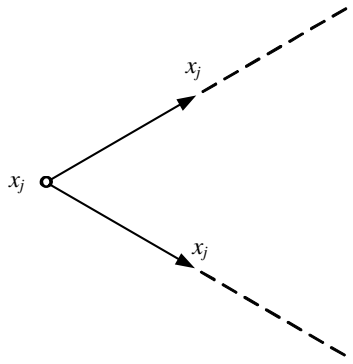


Figure 8-9 - Synaptic divergence or fan-out

4. The state of a *neuron* may be defined in terms of its induced local field or its output signal. A *neuron* is represented by a node on the graph and its activation function, denoted by $\varphi(v)$, can assume three different forms:

1. Threshold function

Mathematically described by:

$$y_k = \begin{cases} 1 & \text{if } v_k \geq 0 \\ 0 & \text{if } v_k < 0 \end{cases} \tag{8-68}$$

where v_k is the induced local field of the neuron, given by:

$$v_k = \sum_{j=1}^{m} w_{kj} x_j + b_k \tag{8-69}$$

This kind of neuron, known as the *McCulloch-Pitts model*, has an output of **1** if the induced local field is nonnegative and **0** otherwise.

2. Piecewise-Linear function

This kind of function can be described by:

$$\varphi(v) = \begin{cases} 1, & v \geq +\frac{1}{2} \\ v, & +\frac{1}{2} > v > -\frac{1}{2} \\ 0, & v \leq -\frac{1}{2} \end{cases} \tag{8-70}$$

where the amplification factor inside the linear region of the operation is assumed to be unity.

3. Sigmoid function

Having an s-shaped graph, this type of function is the most common form of activation function. An example of the sigmoid function is the **logistic function**, defined by:

$$\varphi(v) = \frac{1}{1 + exp(-av)} \tag{8-71}$$

where *a* is the slope parameter of the sigmoid function. By varying it, we obtain sigmoid functions of different slopes.

8.3.5.1 Types of Neural Networks

According to Kröse and van der Smagt [87], Neural Networks can be divided into two essential topologies:

- ***Feed-forward networks***
  "where the data flow from input to output units is strictly feedforward. The data processing can extend over multiple (layers of) units, but no feedback connections are present, that is, connections extending from outputs of units to inputs of units in the same layer or previous layers".

- ***Recurrent networks***
  "that do contain feedback connections. Contrary to feed-forward networks, the dynamical properties of the network are important. In some cases, the activation values of the units undergo a relaxation process such that the network will evolve to a stable state in which these activations do not change anymore. In other applications, the change of the activation values of the output neurons is significant, such that the dynamical behaviour constitutes the output of the network".

8.3.5.1.1 Feed-forward Neural Networks

Within this group of Neural Networks a distinction can also be done between implementations with one or more "neurons" layers.

8.3.5.1.1.1 Single Layer Neural Networks

With a unique layer of "neurons", the **Perceptron**, initially introduced by Frank Rosenblatt [88] constitutes a paradigm for this kind of implementations. Built around the McCulloch-Pitts neuron model, the Perceptron has a signal-flow graph like the one show in the figure below.The inputs x_1, x_2,…, x_m influence the output y according to their respective weights, w_1, w_2,…, w_m. With an externally applied bias, denoted by b, the hard limiter input, v, is then given by:

$$v = \sum_{i=1}^{m} w_i x_i + b \tag{8-72}$$

The objective of the perceptron consists on correctly classifying the input set as belonging to one of two classes. If the result of the above expression is +1 – which happens if a predefined threshold, θ, is achieved - then, the point represented by the perceptron inputs will belong to class C_1. If the result is -1, the point belongs to C_2. The result of the classification should be given by:

$$w^T \mathbf{x} > 0 \quad \text{for every input } \mathbf{x} \text{ belonging to } C_1$$

$$w^T \mathbf{x} < 0 \quad \text{for every input } \mathbf{x} \text{ belonging to } C_2$$

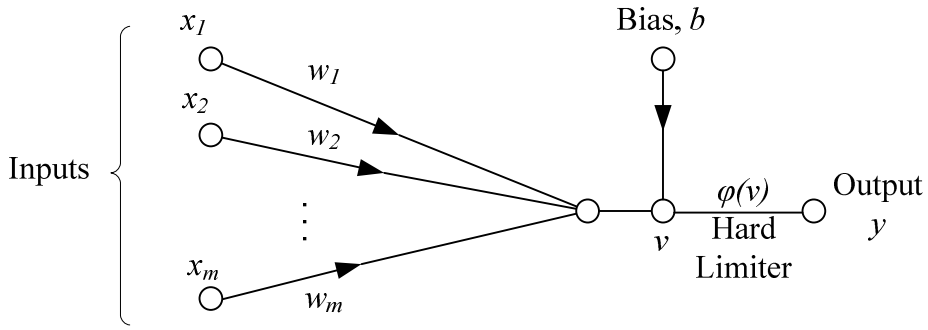


Figure 8-10 – Perceptron signal-flow

To allow for the perceptron learning process, a sample is presented to the network and for each input x_i, a new weight value, w_i, is computed by adding a correction to the old value and the process can be represented by the following equation:

$$w_i(t+1) = w_i(t) + \Delta w_i(t) = w_i(t) + \eta(t)x(t) \tag{8-73}$$

Where $\eta(t)$ is the network learning rate parameter. The learning algorithm can then be described as executing the following actions:

1. Start the execution with random weights for every connection;
2. Select an input vector *x* from the set of training samples;
3. If the perceptron gives an incorrect response, modify all connections weights, w_i, according to the rule above;
4. Go back to step 2.

8.3.5.1.1.2 Multiple Layer Neural Networks

Since there are no hidden neurons within the Single Layer Perceptron above, it can not classify inputs that are not linearly separated. The solution came with the Mutilayer Perceptron. It has a layered structure. Each layer consists of units which receive their input from units from a layer directly below and send their output to units in a layer directly above it. There are no connections between neurons on the same layer. Graphically and assumming one single hidden layer and five different inputs, the network should look like the figure below.

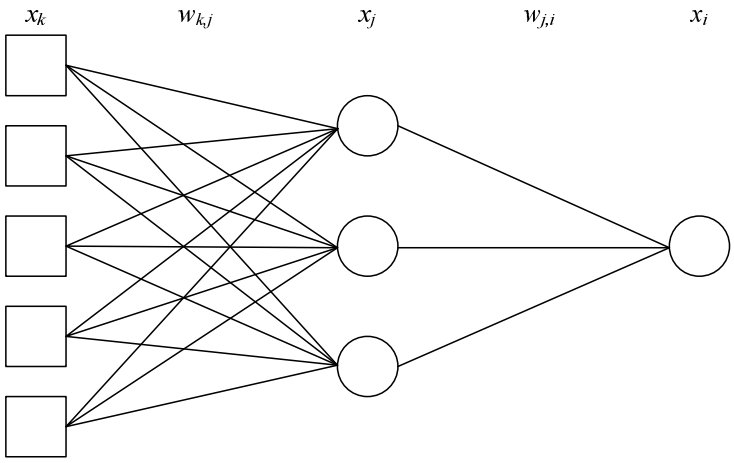


Figure 8-11 - Multilayer Perceptron

Using the same conventions used before - x_k represents the input values, x_j the values at the hidden

layer nodes, x_i the network output and w_{kj} and w_{ji} represent the weights for the connections between the nodes with the same order - within this network, the output of the jth hidden node is given by:

$$x_j = \sum_{i=1}^{m} w_{k,j} x_k \tag{8-74}$$

And the outputs of the network are obtained by applying the same concept to the values calculated for the hidden nodes:

$$x_i = \sum_{i=1}^{n} w_{j,i} x_j \tag{8-75}$$

8.3.5.2 **Recurrent Neural Networks**

When we introduce a cycle on the above described networks, when, for instance, we connect a hidden unit to itself over a weighted connection or connect hidden units to input units, we are building a recurrent network. This type of implementations has signal flow that, for a single-loop feedback system, can be represented by the following figure:

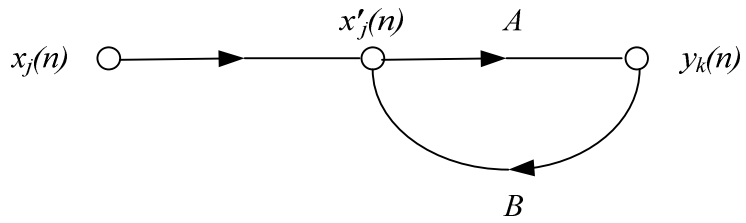


Figure 8-12 - Signal-flow graph for a single-loop feedback system

From here the following relations can be extracted:

$$y_k(n) = A\left[x'_j(n)\right] \tag{8-76}$$

$$x'_j(n) = x_j(n) + B[y_k(n)] \tag{8-77}$$

And, eliminating $x_j'(n)$ we get:

$$y_k(n) = \frac{A}{1 - AB}\left[x_j(n)\right] \tag{8-78}$$

$A/(1 - AB)$ is usually known as the closed-loop operator and AB as the open-loop operator. Assuming that $A = w$, a fixed weight and $B = z^{-1}$, the *unit-delay operator*, the above relation becomes:

$$\frac{A}{1 - AB} = \frac{w}{1 - wz^{-1}} = w(1 - wz^{-1})^{-1} \tag{8-79}$$

this can be rewritten as:

$$\frac{A}{1 - AB} = w\sum_{l=0}^{\infty} w^l z^{-l} \tag{8-80}$$

giving:

$$y_k(n) = w\sum_{l=0}^{\infty} w^l z^{-l}\left[x_j(n)\right] \tag{8-81}$$

Knowing that z^{-1} can be given by:

$$z^{-l}\left[x_j(n)\right] = x_j(n - l) \tag{8-82}$$

where $x_j(n - l)$ is a sample of the input signal delayed by l time units, we can at last express the output signal by:

$$y_k(n) = \sum_{l=0}^{\infty} w^{l+1} x_j(n - l) \tag{8-83}$$

From this, two different situations may arise:

1. $|w| < 1$ and the system becomes stable since $y_k(n)$ is exponentially convergent;

2. $|w| \geq 1$ and the output signal diverges make the system unstable. This divergence will be linear if $|w| = 1$ and exponential otherwise.

In 1982, Hopfield [89] brings together several earlier ideas concerning this type of networks and presents a complete mathematical analysis. The Hopfield network, as it is known, is graphically represented by any of the configurations of figure 8-13 below.

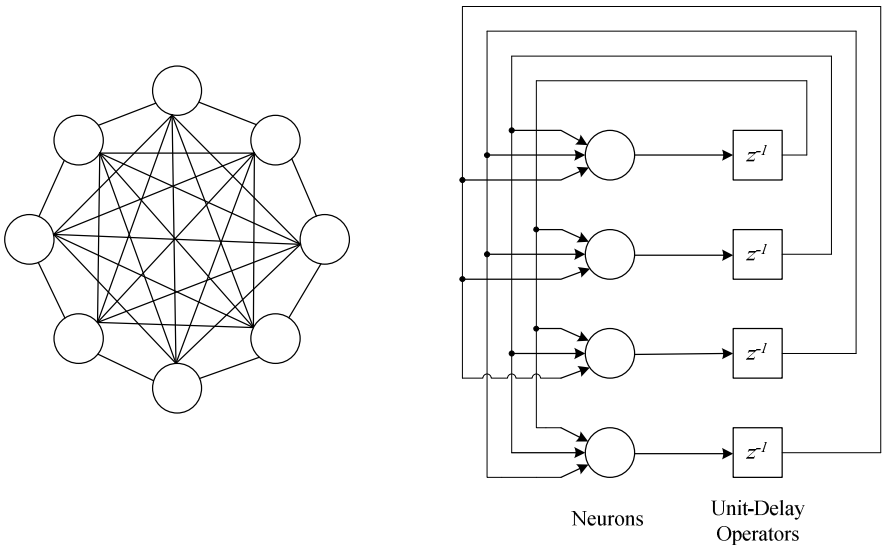


Figure 8-13 - Hopfield Networks (General and Architectural views)

Unlike Perceptron modelling, that required synchronous neurons like a conventional digital computer, this model does not require such synchronism. In fact *there is no evidence for such global synchrony and, given the delays of nerve signal propagation, there would be no way to use global synchrony effectively* [89].

A Hopfield network consists of n totally coupled units that is, each unit is connected to all other units except itself – as can be seen in figure 8-13, above. The network is symmetric because the weight w_{ij} for the connection between units *i* and *j* is equal to the weight w_{ji} of the connection from unit *j* to unit *i*.

The absence of a connection from each unit to itself avoids a permanent feedback of its own state value [90].

A simplified model, with only three nodes, is represented in figure 8-14, and will be used to describe the network principles. Each of the nodes can assume one of two states; 1 or -1.

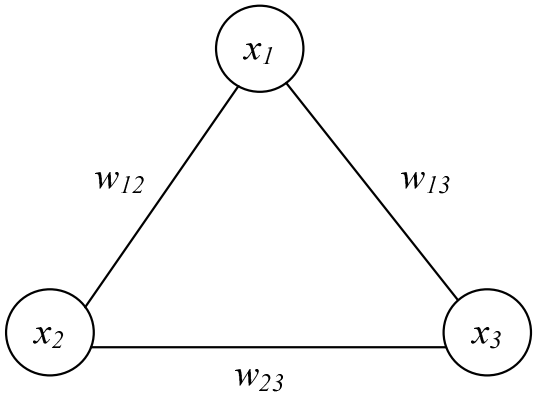


Figure 8-14 - A Hopfield network with 3 neurons

The connections in this network with n nodes can be represented using an $n \times n$ weight matrix $\mathbf{W} = \{w_{ij}\}$ with a zero diagonal. Only with this zero diagonal the network dinamycs leads to stable states. This condition is necessary but may not be sufficient. To guarantee the network's stability, it is also necessary that the weight matrix be symmetric.

The nodes of these networks can be assigned a threshold θ different from zero. So, each node selected for a state update change to state 1 if its total excitation value is greater than θ, and to -1 otherwise. As can be seen, this is similar to the activation rule for perceptrons and so, Hopfield networks can be seen as asynchronous recurrent networks of perceptrons.

The energy $E(\mathrm{x})$ of a state x of the network is then given by:

$$E(\mathrm{x}) = -\frac{1}{2}\sum_{j=1}^{n}\sum_{i=1}^{n} w_{ij}x_i x_j + \sum_{i=1}^{n}\theta_i x_i \qquad (8\text{-}84)$$

It can be shown that it is possible to transform a learning problem in a Hopfield network with n units into a learning problem for a perceptron of dimension $n + n(n-1)/2$, that is, $n(n+1)/2$ [90].
Figure 8-15 below shows an example of a Hopfield network and its transformation into the equivalent perceptron. An iteration of the perceptron learning algorithm updates only the weights of the edges attached to a single unit and its threshold.

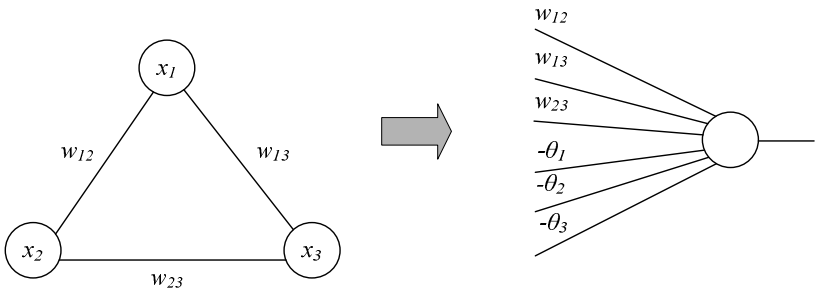


Figure 8-15 - A Hopfield network seen as a Perceptron

If a correction is needed because of the sign of the calculated values for node 1, then only the weights w_{12}, w_{13}, …, w_{1n} and the threshold θ_1 must be updated. This means that it is possible to use perceptron learning or the delta rule locally. During training all units are set to the desired stable states. If the sign of a unit's excitation is incorrect for the desired state, then the weights and threshold of this individual perceptron are corrected in the usual manner.

8.3.5.3 Learning Paradigms

The objective of learning can be achieved with or without a teacher [5]. The first of these paradigms is usually known as Supervised learning.

8.3.5.3.1 Learning with a teacher (Supervised learning)

This learning method can be conceptually described as a situation that corresponds to the existence of an entity, exterior to the network that knows the environment where it is inserted and can analyse the network responses to the various inputs, giving it feedback information as to their correctness. This is usually achieved through the submission of a series of previously correctly classified input vectors. Every bad response originates an adjustment in the weights of the various neural connections, so that next test gives more accurate answers. This is of course an iterative process and should stop when the

neural network is able to give the same responses that would be given by its teacher if facing similar conditions.

Once this level of accuracy is reached, the network can start executing the tasks it had, until then, been trained for. This is also know as error-correction learning and can be depicted by the following figure.

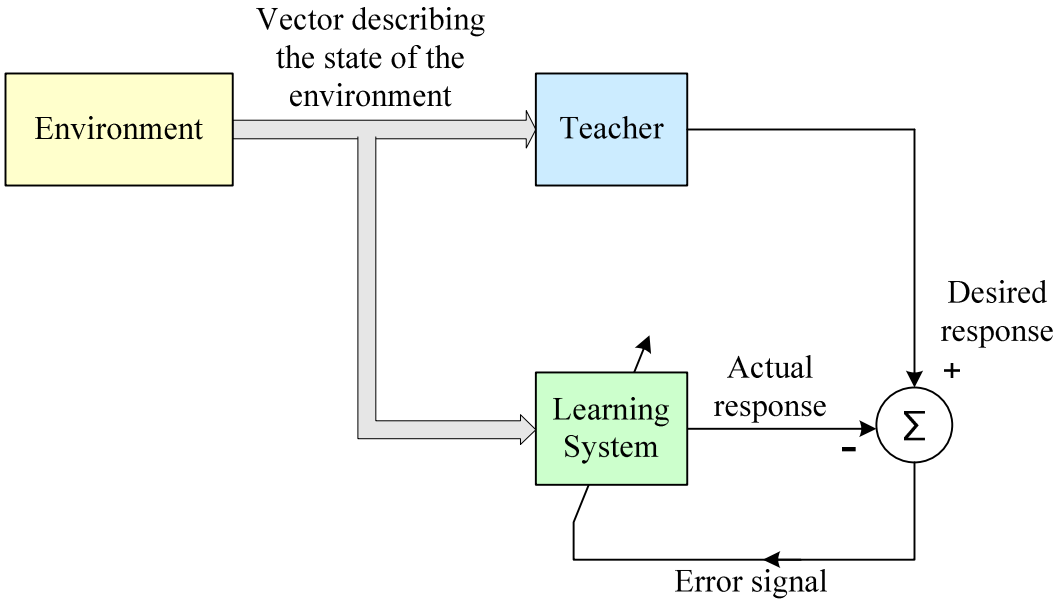


Figure 8-16 - Supervised learning

8.3.5.3.2 Learning without a teacher

When there is no teacher available with enough knowledge to alow for the previous learning method, the network will have to train itself. In this case, two different methods are known:

- **Reinforcement learning**
- **Unsupervised learning**.

8.3.5.3.2.1 Reinforcement learning

According to Sutton and Barto [91], "*Reinforcement learning is learning what to do - how to map situations to actions - so as to maximize a numerical reward signal. The learner is not told which actions to take, as in most forms of machine learning, but instead must discover which actions yield the most reward by trying them. In the most interesting and challenging cases, actions may affect not only the immediate reward but also the next situation and, through that, all subsequent rewards.*
These two characteristics - trial-and-error search and delayed reward - are the two most important distinguishing features of reinforcement learning".

To do this, the network has to analyse a temporal sequence of state vectors which, once processed, will originate some kind of heuristic cost function which is supposed to be minimized by the system. This type of learning system can be represented by the next figure.

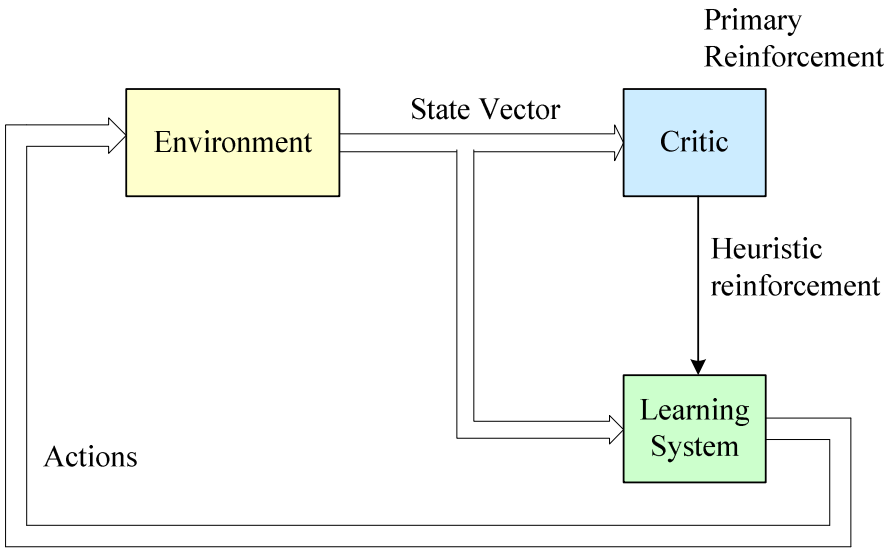


Figure 8-17 - Reinforcement learning

The input vector is presented both to the Learning System and the Critic element. According to the previously defined heuristic, the cost of the previous interactions with the environment are evaluated and new actions are are suggested so that this cost can be reduced.

8.3.5.3.3 Unsupervised learning

In the words of Ghahramani [92] with unsupervised learning, *a machine simply receives inputs x_1, x_2, ..., but obtains neither supervised target outputs, nor rewards from its environment*. Dayan [93] states

that *Unsupervised learning studies how systems can learn to represent particular input patterns in a way that reflects the statistical structure of the overall collection of input patterns. By contrast with SUPERVISED LEARNING or REINFORCEMENT LEARNING, there are no explicit target outputs or environmental evaluations associated with each input; rather the unsupervised learner brings to bear prior biases as to what aspects of the structure of the input should be captured in the output*

Both definitions reflect the fact that with this type of learning, the network will be presented with a set of values – input vectors – and will try to adapt its responses to the statistical regularities of the detected states. Once the network assumes it is tuned, it starts to – automatically - create classes.
This learning paradigm can be represented by the following scheme:

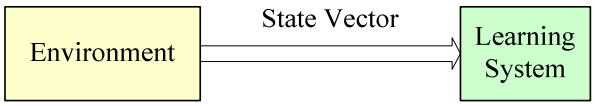
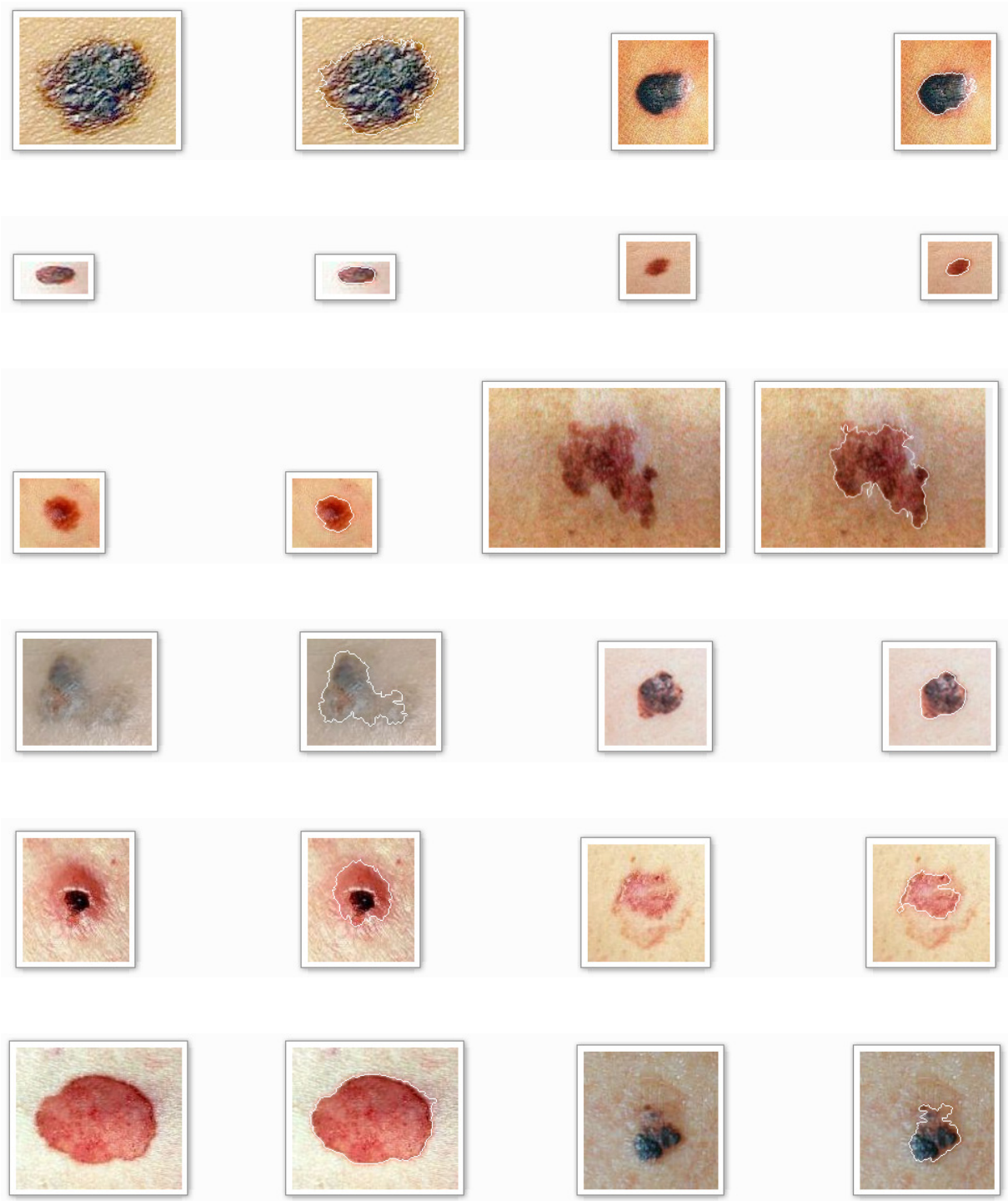
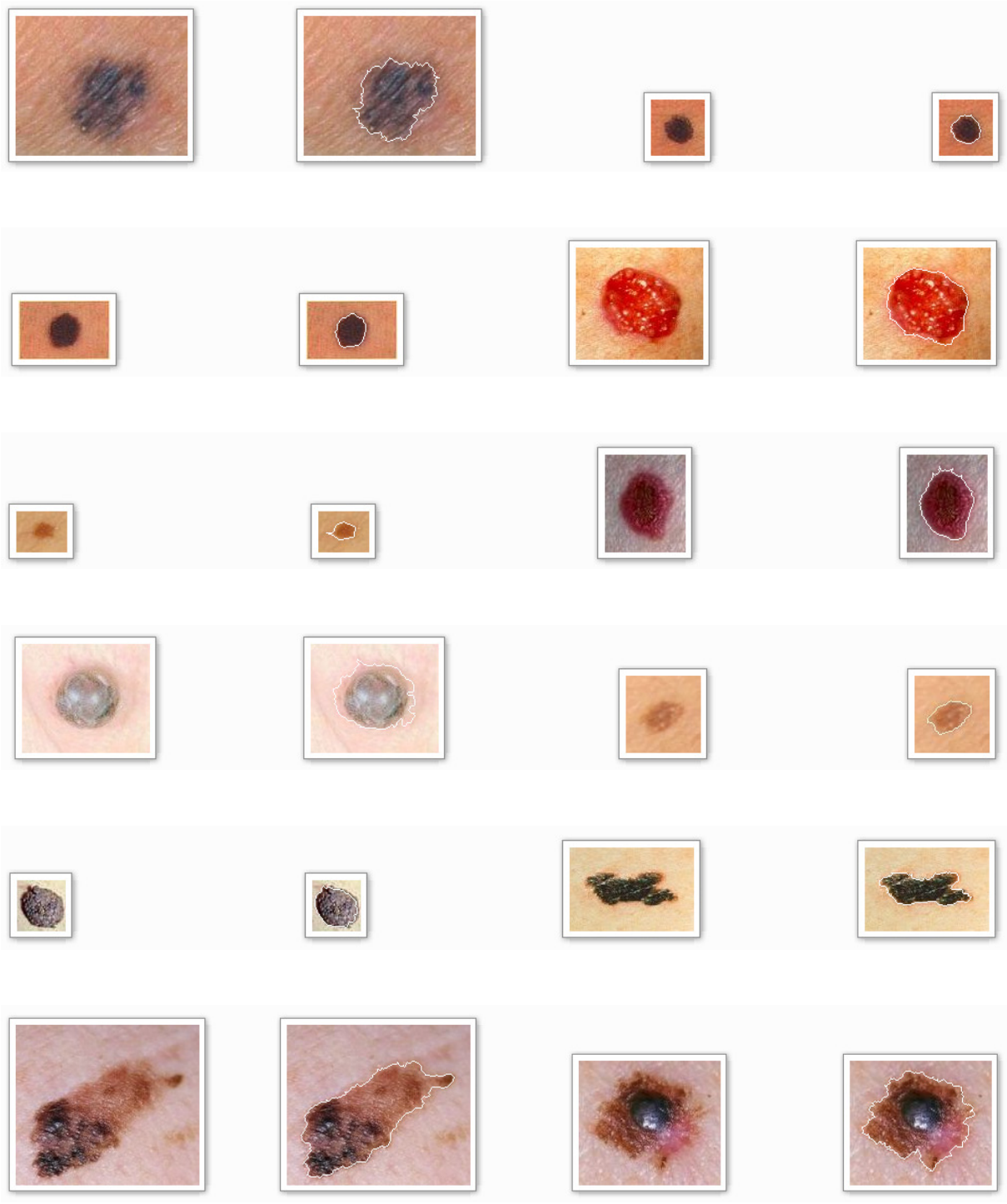
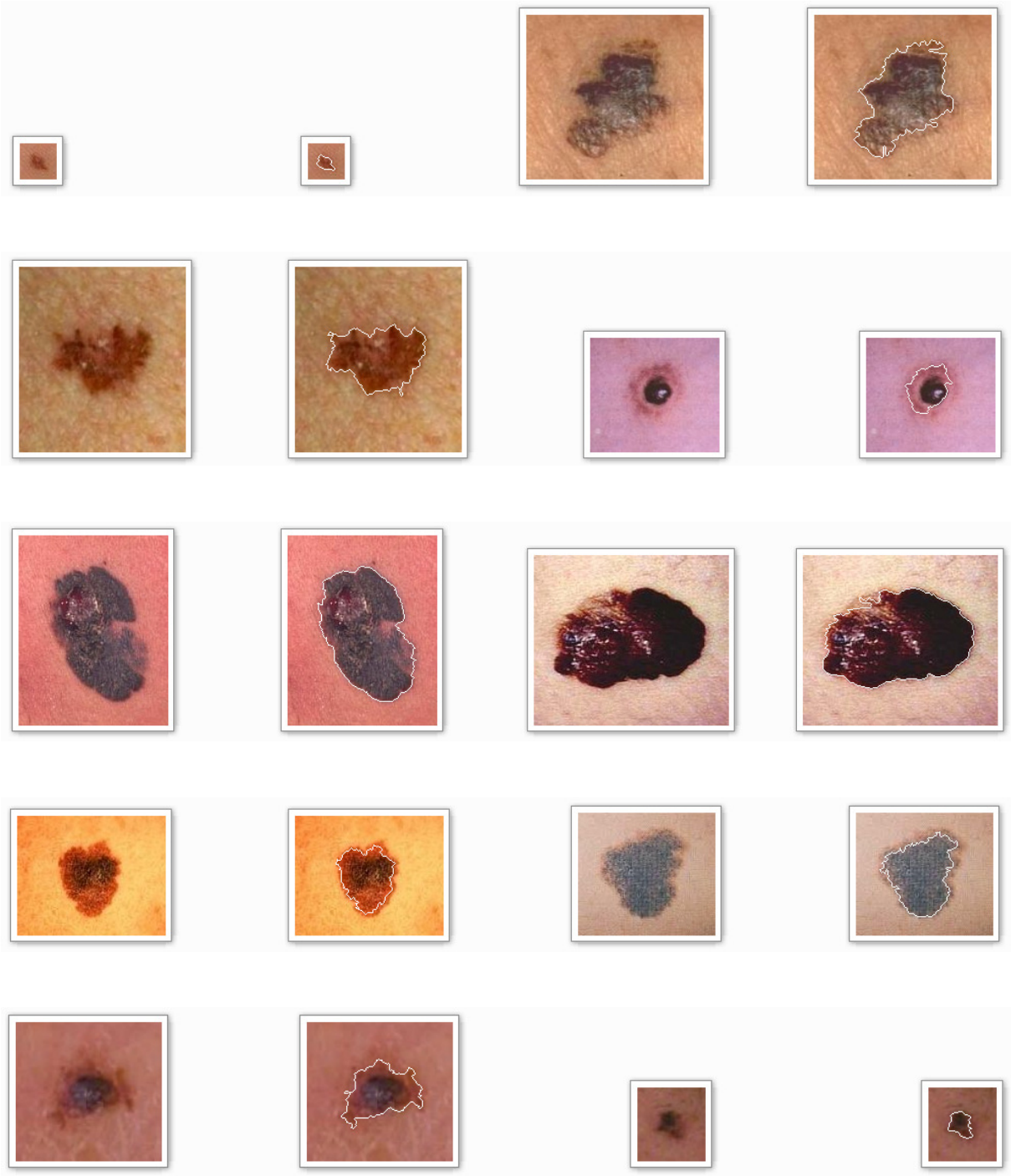


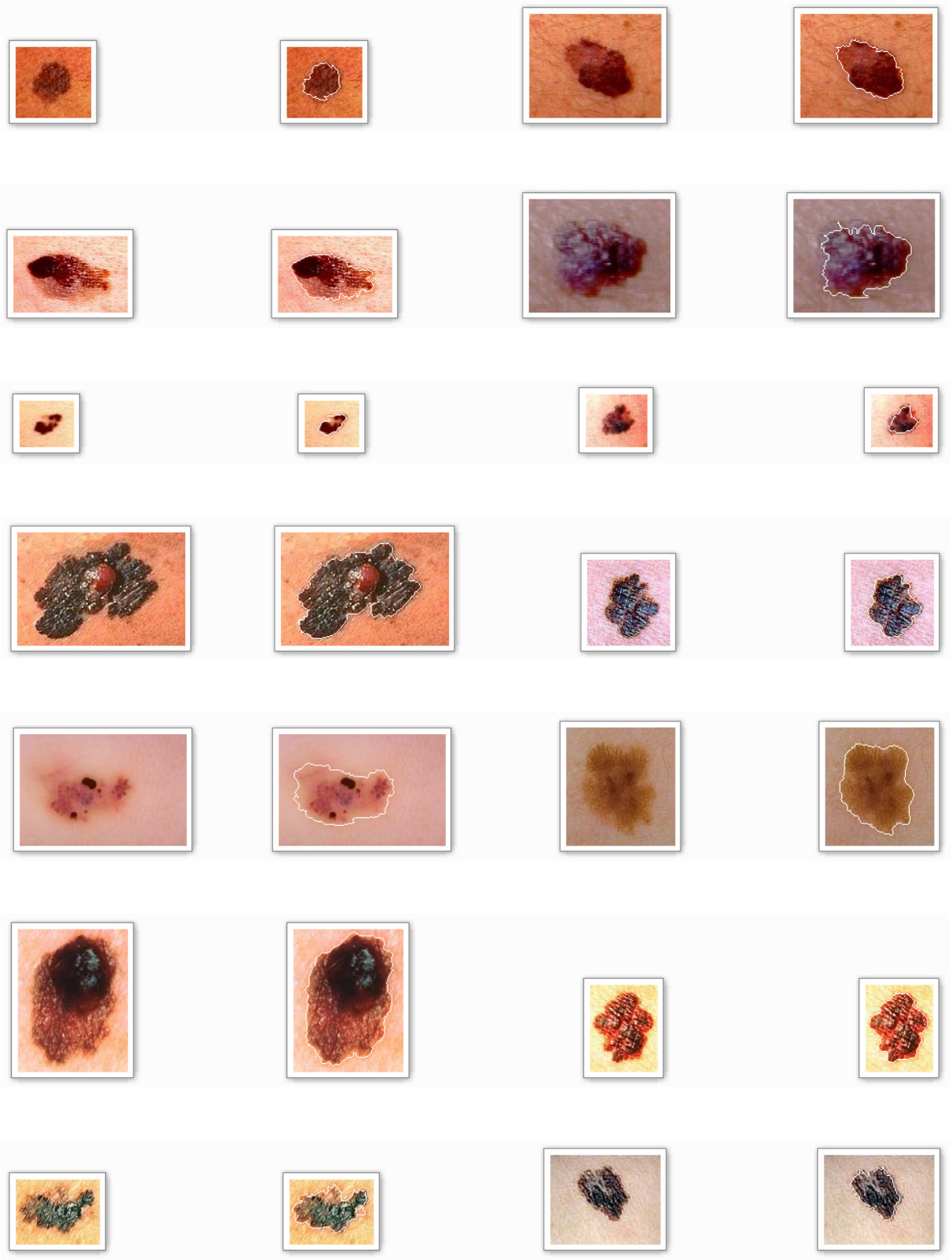
Figure 8-18 - Unsupervised learning

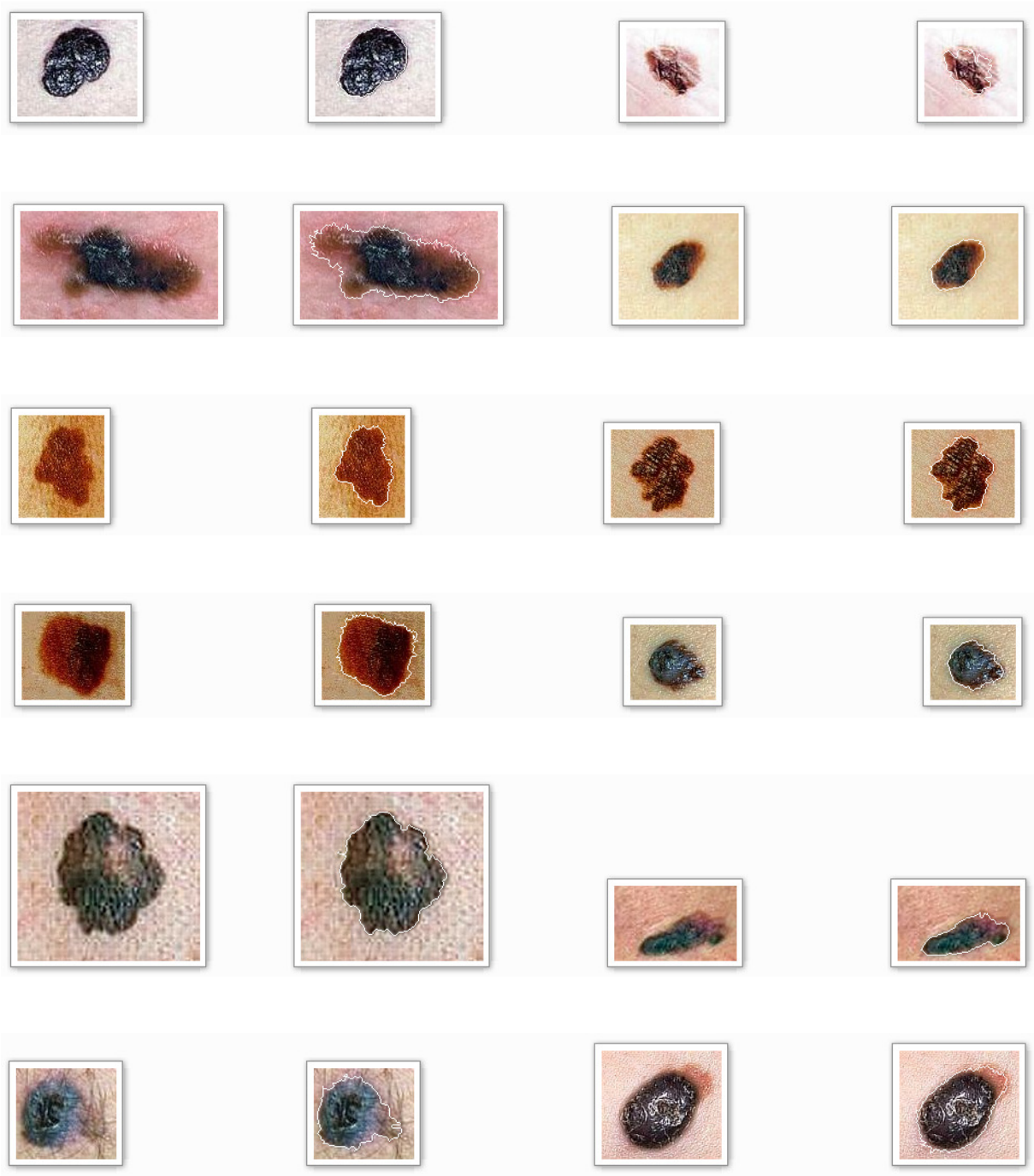
8.4 Appendix D – Image samples with detected edges

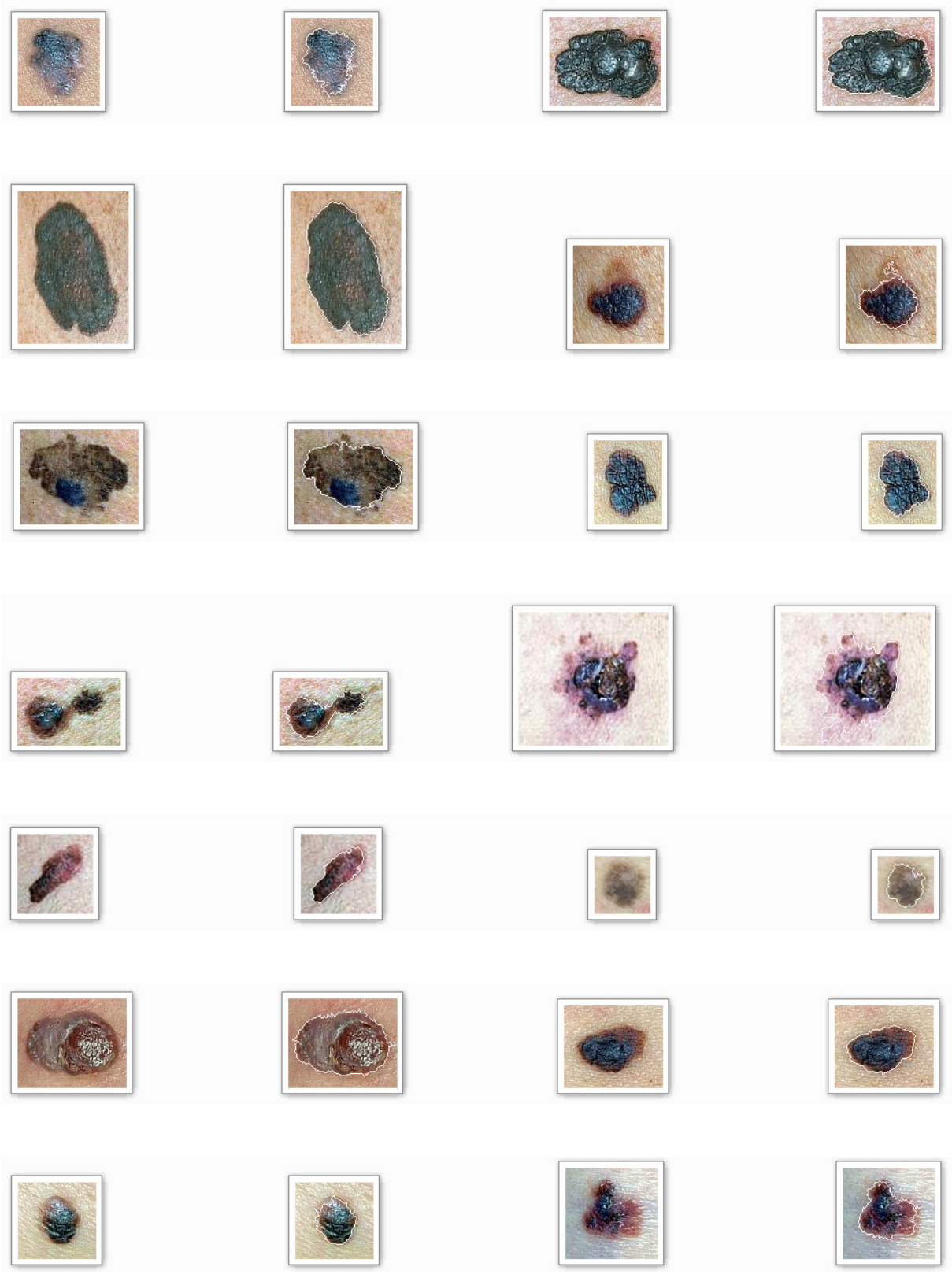


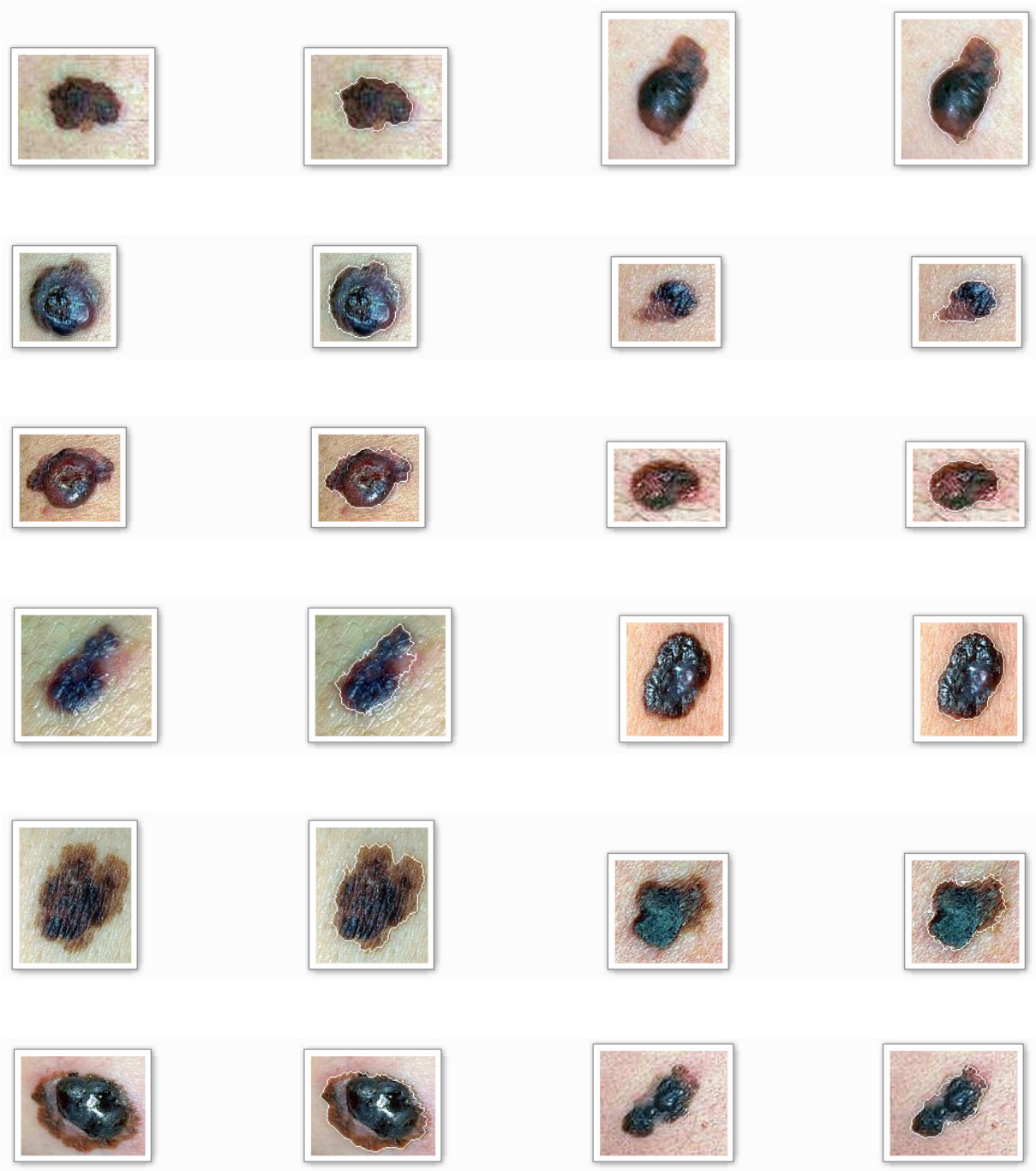


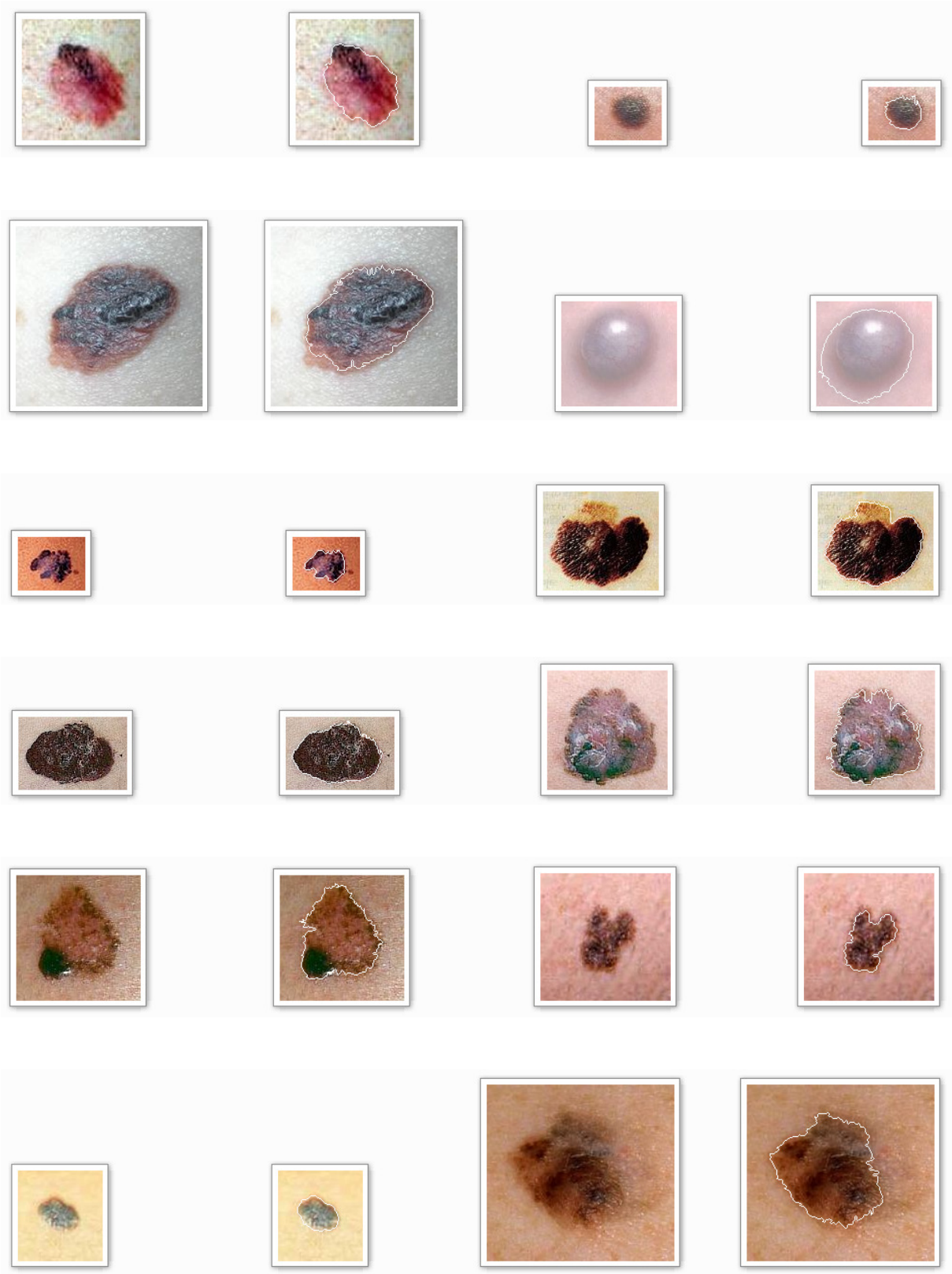


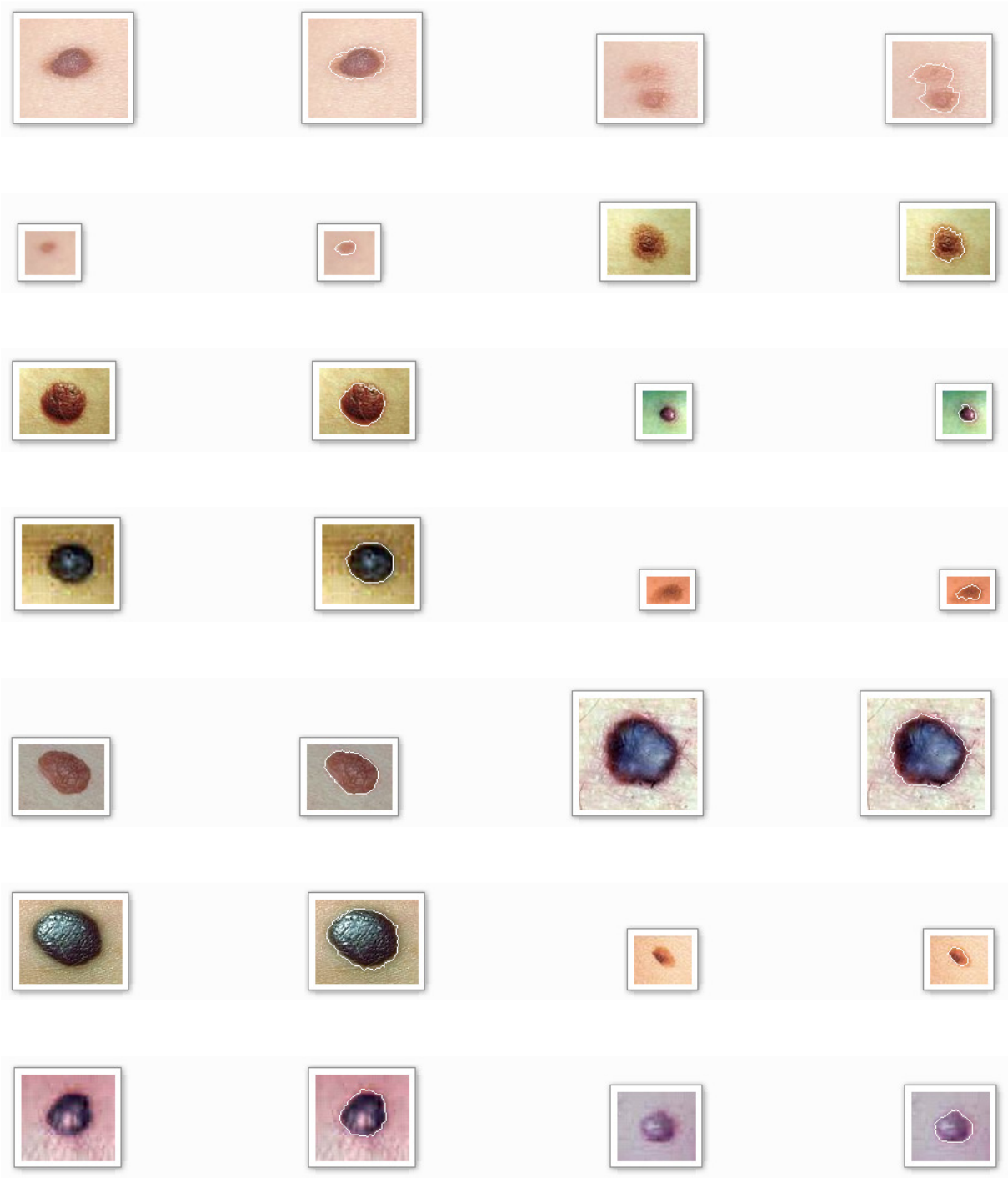


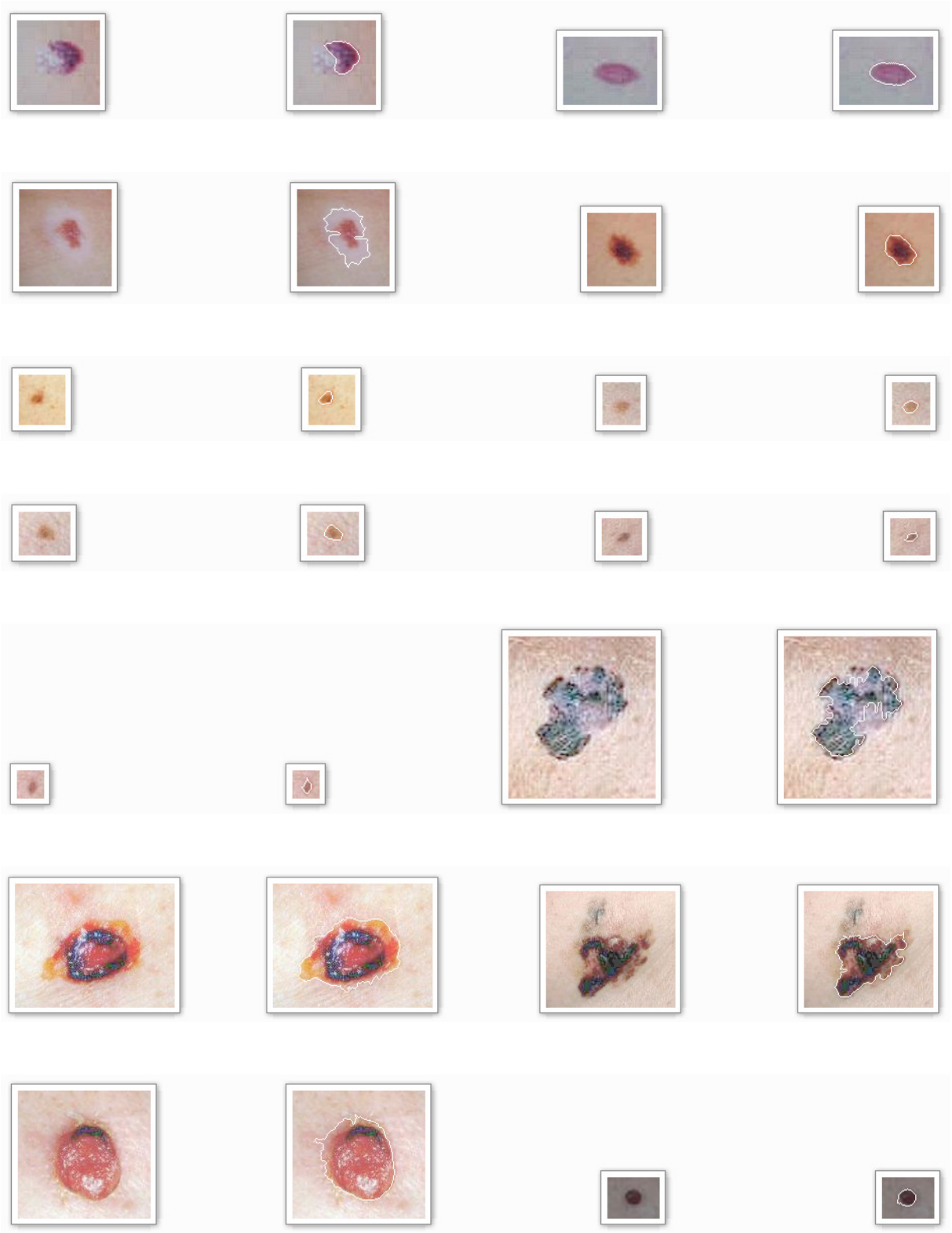


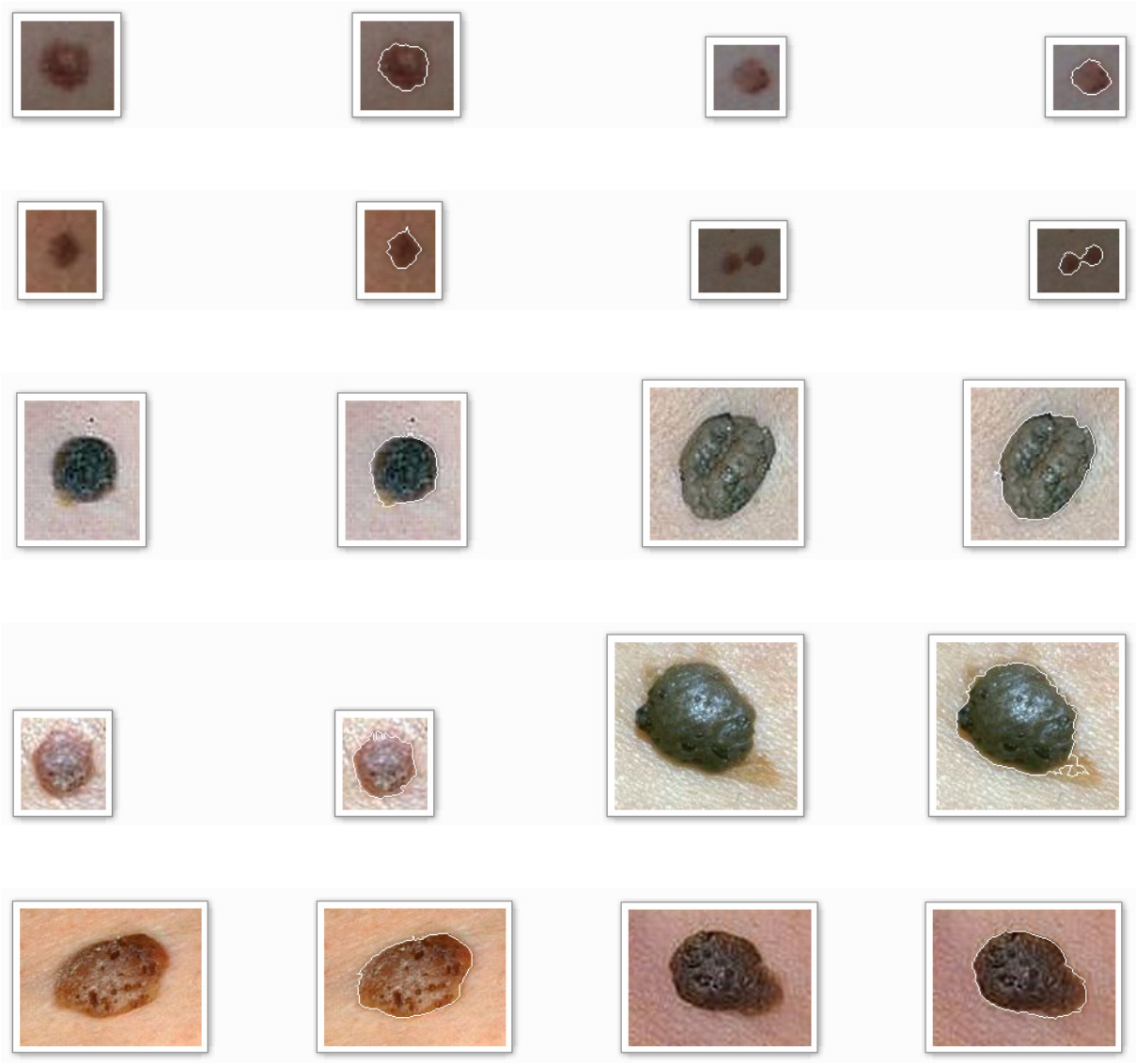












8.5 Appendix E – Features

8.5.1 Extracted

File_Name	Malignant	Type	Threshold (%)	Lesion	Edge		Rays		Colours		Colour Differences			Asymmetry X			Asymmetry Y		
	(Y/N)			Nr. Points	Nr. Points	Dir. Changes	Sigma	Total	Mean	Sigma	Total	Mean	Sigma	Total	Mean	Sigma	Total	Mean	Sigma
basal_fig4_Combi_T8.png	Y	Basal	8%	Y	Y	Y	Y	Y	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Basal_img0019a_Combi_T1.png	Y	Basal	1%	Y	Y	Y	Y	Y	N	Y	Y	N	N	Y	Y	Y	Y	Y	Y
Basalioma_01_Combi_T1.png	Y	Basal	1%	Y	Y	Y	N	Y	N	Y	Y	Y	Y	N	N	Y	Y	N	Y
Basalioma_02_Combi_T1.png	Y	Basal	1%	Y	Y	Y	N	Y	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Basalioma_03_Combi_T4.png	Y	Basal	4%	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Basalioma_04_Combi_T1.png	Y	Basal	1%	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
basalioma_a09f8_Combi_T1.png	Y	Basal	1%	Y	Y	Y	Y	Y	N	Y	Y	Y	Y	Y	N	N	Y	Y	Y
Carcinoma_basal_02_Combi_T8.png	Y	Basal	8%	Y	Y	Y	Y	Y	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
1287melanoma2_Combi_T9.png	Y	Melanoma	9%	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
7melanoma_Combi_T1.png	Y	Melanoma	1%	Y	Y	Y	Y	Y	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
image_a_Combi_T1.png	Y	Melanoma	1%	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	Y	Y	Y	Y	Y	Y
malig2_Combi_T1.png	Y	Melanoma	1%	Y	Y	Y	Y	Y	N	Y	Y	Y	Y	Y	Y	Y	Y	N	Y
malignant_melanoma_1_Combi_T1.png	Y	Melanoma	1%	Y	Y	Y	Y	Y	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
malignant_melanoma_2_Combi_T6.png	Y	Melanoma	6%	Y	Y	Y	Y	Y	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Melanoma_04_Combi_T6.png	Y	Melanoma	6%	Y	Y	Y	Y	Y	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Melanoma_005_Combi_T6.png	Y	Melanoma	6%	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Melanoma_005a_Combi_T3.png	Y	Melanoma	3%	Y	Y	Y	Y	Y	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Melanoma_006_Combi_T1.png	Y	Melanoma	1%	N	N	N	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
melanoma_007_Combi_T10.png	Y	Melanoma	10%	N	N	N	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
melanoma_009_Combi_T1.png	Y	Melanoma	1%	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

File Name	Malignant (Y/N)	Type	Threshold (%)	Lesion	Edge		Rays	Colours			Colour Differences			Asymmetry X			Asymmetry Y		
				Nr. Points	Nr. Points	Dir. Changes	Sigma	Total	Mean	Sigma	Total	Mean	Sigma	Total	Mean	Sigma	Total	Mean	Sigma
melanoma_01_Combi_T6.png	Y	Melanoma	6%	427	77	36	2,204	147259	187	37,453	50496	8	8,602	102	2	2,076	327	3	2,513
melanoma_012_Combi_T1.png	Y	Melanoma	1%	5085	301	76	9,895	3760070	429	92,862	2747591	12	18,055	8321	14	12,616	2659	8	5,27
melanoma_014_Combi_T1.png	Y	Melanoma	1%	5715	285	119	5,319	3646463	300	64,603	2974751	17	15,906	3313	9	7,244	4641	10	8,285
Melanoma_015_Combi_T1.png	Y	Melanoma	1%	8986	351	118	8,901	940539749	496	156,123	1051327088	19	23,875	495945	16	12,268	835210	22	16,696
Melanoma_016_Combi_T1.png	Y	Melanoma	1%	2726	193	63	3,889	183177298	450	194,903	2040172657	64	72,45	112926	10	7,249	184447	13	9,478
Melanoma_017_Combi_T1.png	Y	Melanoma	1%	2322	235	90	6,697	63519524	439	156,283	803812313	68	76,466	124449	14	9,664	53289	10	6,715
melanoma_018_Combi_T1.png	Y	Melanoma	1%	1626	159	70	4,149	865844	223	151,658	22728490	91	78,243	932	7	4,448	1206	8	4,577
melanoma_019_Combi_T1.png	Y	Melanoma	1%	2967	195	71	4,19	3063825	360	256,615	87595734	93	103,533	2939	12	8,584	2681	10	8,147
Melanoma_02_Combi_T10.png	Y	Melanoma	10%	1035	119	45	1,994	484551	234	93,787	1814351	32	30,61	768	6	7,895	637	4	5,101
melanoma_020_Combi_T1.png	Y	Melanoma	1%	1261	188	59	7,295	6546390	451	213,236	80625162	70	76,753	7855	12	9,832	7874	11	7,408
Melanoma_021_Combi_T3.png	Y	Melanoma	3%	8502	494	210	17,95	4579831	263	125,492	29105181	35	41,593	41099	26	19,663	6379	13	9,896
melanoma_10_Combi_T1.png	Y	Melanoma	1%	2415	184	80	3,534	81621794	575	154,295	672400470	55	70,314	46282	12	9,193	62235	14	10,321
melanoma_a09f2_Combi_T1.png	Y	Melanoma	1%	1853	145	47	4,427	1212980	362	207,551	2982776	29	30,48	3218	18	14,213	1294	12	9,055
melanoma_abdc_01_Combi_T12.png	Y	Melanoma	12%	2700	212	85	5,632	1952998	235	117,266	24366717	67	55,471	1355	7	6,539	3150	10	8,1
melanoma_abdc_02_Combi_T1.png	Y	Melanoma	1%	2657	196	81	3,694	1371874	202	144,411	58881683	101	95,231	650	4	3,125	1105	4	3,834
melanoma_abdc_03_Combi_T2.png	Y	Melanoma	2%	4014	256	102	3,09	1584949	186	104,985	18855614	51	48,26	2088	8	6,354	2612	7	4,622
Melanoma_img0002_Combi_T6.png	Y	Melanoma	6%	1630	137	58	2,475	688791	259	157,638	20557132	89	90,172	599	5	4,273	604	5	7,163
Melanoma_img0003_Combi_T3.png	Y	Melanoma	3%	8468	350	134	5,065	7721001	331	180,863	16589659	28	27,037	5918	12	11,458	6645	11	12,199
Melanoma_img0004_Combi_T1.png	Y	Melanoma	1%	2075	193	70	9,666	599271	242	104,626	1783356	32	27,55	4183	13	9,937	704	6	5,861
Melanoma_img0005a_Combi_T1.png	Y	Melanoma	1%	3688	260	113	5,36	2273087	299	103,126	8370475	35	33,853	3325	10	7,089	1482	6	5,324
Melanoma_img0005b_Combi_T1.png	Y	Melanoma	1%	4453	252	110	6,876	144770555	534	177,649	1052696830	57	63,561	159683	17	12,329	160847	17	11,903
Melanoma_img0010_Combi_T6.png	Y	Melanoma	6%	2436	216	4,762	8,124	6394151	425	128,675	98152709	88	82,789	3438	9	7,779	5251	11	7,947
Melanoma_img0013_Combi_T3.png	Y	Melanoma	3%	5731	282	118	7,558	8249409	359	186,42	193205163	104	93,386	17130	14	11,787	7887	11	9,497
Melanoma_img0014_Combi_T1.png	Y	Melanoma	1%	7999	329	139	11,275	8212668	333	123,237	31644345	37	36,304	2982	13	8,439	9570	15	12,723
Melanoma_img0019_Combi_T7.png	Y	Melanoma	7%	2268	223	98	5,812	2207332	267	157,631	27024060	64	58,541	2075	9	8,821	3088	10	6,882

File Name	Malignant (Y/N)	Type	Threshold (%)	Lesion	Edge		Rays	Colours			Colour Differences			Asymmetry X			Asymmetry Y		
				Nr. Points	Nr. Points	Dir. Changes	Sigma	Total	Mean	Sigma	Total	Mean	Sigma	Total	Mean	Sigma	Total	Mean	Sigma
Melanoma_img0020_Combi_T7.png	Y	Melanoma	7%	4641	283	6,519	8,516	1853584	230	121,611	22025092	59	53,104	6720	11	8,946	1600	7	7,624
Melanoma_img0021_Combi_T1.png	Y	Melanoma	1%	2290	182	4,93	6,035	2812605	331	171,175	67498565	104	91,132	1934	8	5,493	2444	9	8,167
Melanoma_img0024_Combi_T1.png	Y	Melanoma	1%	2170	230	103	11,5	3508617	382	198,052	40531637	69	68,132	9122	16	11,337	4917	12	8,534
Melanoma_img0030_Combi_T2.png	Y	Melanoma	2%	6461	415	152	7,64	40044864	544	181,167	91710821	29	35,763	35789	21	16,444	32837	19	15,756
Melanoma_img0033_Combi_T1.png	Y	Melanoma	1%	1357	145	50	7,195	506977	256	124,346	2089531	36	33,69	808	12	6,615	1177	11	7,282
Melanoma_img0048_Combi_T1.png	Y	Melanoma	1%	1142	127	56	2,152	766214	359	63,292	589777	19	17,321	379	3	2,455	673	4	2,791
Melanoma_img0054_Combi_T3.png	Y	Melanoma	3%	5066	272	104	6,348	2992269	316	118,666	75618947	80	91,011	4670	9	6,788	1997	7	7,264
Melanoma_img0056_Combi_T1.png	Y	Melanoma	1%	2433	178	71	4,901	1473897	311	170,72	14051752	68	55,534	2735	9	8,171	1728	9	9,693
Melanoma_img0056a_Combi_T1.png	Y	Melanoma	1%	1353	142	58	2,36	2009986	399	204,983	29800009	81	79,126	2449	10	10,436	2040	8	6,807
Melanoma_img0064a_Combi_T3.png	Y	Melanoma	3%	2261	202	82	4,333	2786624	365	183,949	32934681	68	67,454	2170	8	8,025	3021	10	8,177
Melanoma_img0081_Combi_T1.png	Y	Melanoma	1%	2671	190	67	4,231	838074	237	120,842	1635400	24	21,977	2295	7	7,563	669	5	4,501
Melanoma_img0085a_Combi_T1.png	Y	Melanoma	1%	4164	230	70	8,419	2887920	304	158,57	6762026	25	27,092	1984	15	11,655	4159	17	10,1
Melanoma_img0090_Combi_T5.png	Y	Melanoma	5%	3520	222	98	1,954	2968828	333	160,672	36966337	63	66,106	2364	8	8,945	2426	6	5,43
Melanoma_img0090a_Combi_T1.png	Y	Melanoma	1%	1555	157	51	5,227	785574	309	163,336	17484340	87	85,317	4580	11	7,785	552	4	4,476
Melanoma_img0092_Combi_T1.png	Y	Melanoma	1%	3088	199	74	4,875	4356233	390	188,843	46316741	70	65,666	2920	8	6,248	3235	10	8,724
Melanoma_img0095a_Combi_T5.png	Y	Melanoma	5%	2414	178	61	4,345	2490688	359	179,467	13494942	48	45,288	3905	11	11,066	1442	5	4,848
Melanoma_img0095b_Combi_T5.png	Y	Melanoma	5%	3505	247	99	8,796	4304110	334	164,721	32722050	49	51,303	4770	18	11,911	5712	18	11,022
Melanoma_img0097_Combi_T1.png	Y	Melanoma	1%	3823	216	86	5,735	142933032	564	124,938	482982944	34	44,486	53103	12	9,254	120531	17	11,661
Melanoma_img0100_Combi_T1.png	Y	Melanoma	1%	5795	304	114	5,789	3567194	271	127,337	37241927	61	54,056	3302	10	9,406	3982	11	9,001
Melanoma_img0100a_Combi_T8.png	Y	Melanoma	8%	3631	269	104	5,373	10220838	379	181,836	121702207	82	68,359	11664	14	10,578	8920	13	9,663
Melanoma_img0102_Combi_T1.png	Y	Melanoma	1%	5630	269	108	5,645	2430376	263	150,071	29563235	51	57,697	5219	13	9,859	3428	12	9,342
Melanoma_img0103a_Combi_T1.png	Y	Melanoma	1%	2607	214	89	8,68	2240211	325	192,686	17973887	51	52,019	4411	19	14,107	3158	15	11,027
Melanoma_img0105_Combi_T1.png	Y	Melanoma	1%	4288	231	88	4,816	3309249	391	154,988	3601531	23	21,095	4746	16	13,193	3513	11	9,254
melanoma_mal_Combi_T4.png	Y	Melanoma	4%	976	123	51	1,559	612352	332	88,148	3207662	43	43,209	458	3	3,391	632	3	3,937
Melanoma_malignant_Combi_T1.png	Y	Melanoma	1%	10310	439	159	9,164	8098275	350	86,723	50961125	52	47,497	12783	17	11,497	6033	13	10,593

File Name	Malignant (Y/N)	Type	Threshold (%)	Lesion	Edge		Rays	Colours			Colour Differences			Asymmetry X			Asymmetry Y		
				Nr. Points	Nr. Points	Dir. Changes	Sigma	Total	Mean	Sigma	Total	Mean	Sigma	Total	Mean	Sigma	Total	Mean	Sigma
melanoma_nodule_Combi_T1.png	Y	Melanoma	1%	7557	311	124	3,549	7735757	536	52,435	659647	5	6,782	3630	10	7,222	4072	9	6,378
melanoma_palpabile_Combi_T1.png	Y	Melanoma	1%	935	113	41	3,418	629575	315	164,724	4504655	50	49,528	2727	9	6,674	683	4	4,627
Melanoma_XX01A_1_Combi_T1.png	Y	Melanoma	1%	5734	276	108	4,813	2381785	197	171,621	127200095	96	104,168	3380	6	4,448	2290	6	4,106
Melanoma_XX04_Combi_T1.png	Y	Melanoma	1%	4165	243	100	6,158	1051980	179	129,529	66723685	113	108,517	1603	4	4,918	1216	7	4,353
melanoma-1_Combi_T4.png	Y	Melanoma	4%	3974	285	128	6,116	4241796	337	53,247	10120633	34	28,914	2111	8	8,105	4698	10	8,188
melanoma10_Combi_T9.png	Y	Melanoma	9%	9549	377	154	17,244	8862529	314	90,299	12898264	20	21,679	8050	19	17,645	10415	19	17,583
Melanoma1707_Combi_T6.png	Y	Melanoma	6%	11550	444	159	16,509	57101345	418	229,827	329081232	36	49,79	112644	24	17,31	71799	21	14,55
melanoma-2_Combi_T10.png	Y	Melanoma	10%	2845	254	91	6,532	1773839	255	44,805	460089	9	8,367	3146	9	7,778	2886	9	7,021
melanoma3_Combi_T1.png	Y	Melanoma	1%	6327	411	158	9,625	8901414	300	65,993	11863061	22	20,273	6517	17	11,262	12302	17	10,783
Melanoma3200_Combi_T9.png	Y	Melanoma	9%	2630	210	82	3,895	514385651	465	117,373	498310286	16	21,587	702685	22	15,816	570152	19	13,564
melanoma4_Combi_T6.png	Y	Melanoma	6%	4704	315	122	7,785	2348423	233	60,741	1197981	12	11,136	9815	14	11,169	3777	10	6,671
melanoma8_Combi_T10.png	Y	Melanoma	10%	1487	147	55	3,039	1031278	331	103,364	1852981	21	25,159	753	6	4,214	1329	8	4,529
melanoma-fig1_Combi_T2.pn	Y	Melanoma	2%	6726	396	164	4,865	12996554	417	144,375	92613023	53	55,39	6600	10	8,148	11773	14	14,197
melanoma-fig2a_Combi_T1.pn	Y	Melanoma	1%	5860	352	152	7,002	5168197	251	98,224	24977192	32	35,496	4575	8	7,483	7454	14	11,903
melanoma-fig3_Combi_T1.pn	Y	Melanoma	1%	2249	211	81	5,791	2649877	311	156,255	5703351	29	26,42	2794	10	9,772	4461	15	14,517
melanoma-fig4_Combi_T3.pn	Y	Melanoma	3%	1204	114	43	2,777	15796652	571	102,602	13363644	17	22,428	20040	15	11,31	14247	13	9,649
n138a_Combi_T1.png	Y	Melanoma	1%	9464	418	196	6,167	5790966	228	79,303	13763909	24	23,558	4907	13	10,138	6885	13	11,661
Escamosas_img0046_Combi_T1.png	Y	Squamuous	1%	2312	172	72	3,493	1071924	222	78,382	3188345	25	26,192	349	2	3,255	667	3	2,769
Atypical_mole_001_Combi_T6.png	N	Atypical	6%	269	54	18	1,816	66268	272	66,656	130685	28	24,637	192	4	2,231	77	2	2,048
Atypical_mole_002_Combi_T4.png	N	Atypical	4%	1046	105	46	1,636	431017	314	86,21	658539	26	22,672	334	4	1,648	306	3	3,012
keratosis1_Combi_T1.png	N	Keratosis	1%	1174	112	40	1,645	528626	329	136,854	5941296	68	62,554	475	4	2,97	410	4	2,341
Queratose_img0088_Combi_T4.png	N	Keratosis	4%	2864	180	62	1,657	1218076	250	100,915	2267407	22	22,068	880	4	4,087	605	2	2,308
Queratose_img0093_Combi_T1.png	N	Keratosis	1%	5682	256	82	4,247	2406303	288	63,129	5626451	25	26,287	2513	8	5,517	2262	10	7,858
Queratose_img0094_Combi_T1.png	N	Keratosis	1%	2372	187	75	1,98	5412403	463	126,247	7121234	26	25,159	4404	9	6,805	5067	10	8,81
Queratose_img0095_Combi_T12.png	N	Keratosis	12%	7547	358	114	7,2	5285449	301	133,989	12410018	25	27,111	7912	17	14,745	4741	11	10,762

File Name	Malignant (Y/N)	Type	Threshold (%)	Lesion	Edge		Rays	Colours			Colour Differences			Asymmetry X			Asymmetry Y		
				Nr. Points	Nr. Points	Dir. Changes	Sigma	Total	Mean	Sigma	Total	Mean	Sigma	Total	Mean	Sigma	Total	Mean	Sigma
seborrheic_keratosis_02_Combi_T1.png	N	Keratosis	1%	5848	271	94	7,123	2870741	334	112,141	4472992	25	23,173	4029	12	8,801	2886	12	10,451
seborrheic_keratosis_03_Combi_T4.png	N	Keratosis	4%	5982	266	105	6,692	1401030	195	58,956	2795427	21	20	5618	16	11,384	1393	9	7,976
Atypical nevus_01_Combi_T1.png	N	Nevus	1%	477	77	18	3,443	757199	480	176,896	5121659	49	61,612	2843	5	3,747	581	2	2,377
Benign nevus_01_Combi_T4.png	N	Nevus	4%	4354	277	118	4,725	3155894	309	58,056	3668294	23	19,391	2452	8	5,333	3737	10	7,435
Benign nevus_02_Combi_T1.png	N	Nevus	1%	600	77	31	1,339	134523	202	56,025	259703	22	20,688	111	1	1,708	124	2	1,815
Benign nevus_03_Combi_T11.png	N	Nevus	11%	849	87	30	1,3	3454350	353	125,053	3641181	19	19,672	79933	14	10,256	24078	9	6,816
nevi4_Combi_T1.png	N	Nevus	1%	731	99	46	1,243	403735	312	88,847	1908521	42	40,162	189	2	1,887	298	3	2,208
nevi4a_Combi_T1.png	N	Nevus	1%	1231	113	46	1,487	321690	218	83,104	2496694	44	42,19	210	2	1,877	159	1	1,651
nevo_03_Combi_T3.png	N	Nevus	3%	210	44	16	1,279	57336	261	133,521	559206	52	54,102	54	1	1,306	62	2	1,549
nevo_04_Combi_T1.png	N	Nevus	1%	1292	115	38	1,73	201990	142	72,878	507696	23	19,339	473	3	1,93	217	1	1,778
nevo_congenito_Combi_T1.png	N	Nevus	1%	244	52	19	2,546	60192	330	33,915	23893	15	12,288	178	4	2,204	25	1	0,929
nevo_img0030_Combi_T1.png	N	Nevus	1%	1593	127	38	3,281	491394	297	49,248	464510	19	17,117	1096	7	4,239	508	6	3,709
Nevo_img0031_Combi_T10.png	N	Nevus	10%	3937	219	82	3,033	2779745	336	167,196	4187378	24	22,935	3287	8	7,907	1461	5	6,124
nevo_img0084_Combi_T2.png	N	Nevus	2%	2843	182	67	2,906	1026445	253	151,305	28442254	72	85,948	1201	6	7,528	934	5	4,546
nevo_lentigginoso_Combi_T1.png	N	Nevus	1%	246	46	18	1,905	3276938	586	100,846	2299815	18	21,354	2685	10	6,941	2852	6	4,486
nevoa3_small_Combi_T1.png	N	Nevus	1%	1469	125	46	1,891	563007	263	137,71	893468	22	20,833	435	3	3,169	412	4	5,27
nevoc2_small_Combi_T1.png	N	Nevus	1%	847	92	34	2,022	331069	388	77,587	272362	17	18,52	260	2	1,751	204	3	1,958
nevodis_small_Combi_T0.png	N	Nevus	0%	757	93	27	3,538	397863	379	78,944	196796	18	14,177	467	4	3,025	253	4	4,989
nevosp1_small_Combi_T1.png	N	Nevus	1%	796	96	40	3,99	269404	396	25,055	63611	12	10	837	5	3,155	78	1	1,319
nevosu_small1_Combi_T3.png	N	Nevus	3%	1690	195	63	6,294	2818504	510	52,112	508285	9	9,849	1381	6	4,429	2963	11	7,679
nevou5_small_Combi_T1.png	N	Nevus	1%	671	81	25	2,304	211186	302	68,858	110618	17	13,038	358	5	2,485	303	4	2,785
Nevus_003_Combi_T1.png	N	Nevus	1%	134	34	6	1,465	50632	418	54,627	38601	20	19,545	30	2	1,69	30	1	1,433
Nevus_img0085_Combi_T4.png	N	Nevus	4%	151	35	16	1,331	51717	465	38,802	30670	17	18,055	45	1	1,497	25	1	0,97
Nevus_img0085b_Combi_T1.png	N	Nevus	1%	214	43	14	1,229	75802	456	44,067	37098	15	15,937	48	2	1,924	34	2	1,715
Nevus_img0103_Combi_T4.png	N	Nevus	4%	80	28	10	1,195	19073	389	32,758	9349	14	15,875	42	2	1,374	16	1	1,095

File Name	Malignant (Y/N)	Type	Threshold (%)	Lesion	Edge		Rays	Colours			Colour Differences			Asymmetry X			Asymmetry Y		
				Nr. Points	Nr. Points	Dir. Changes	Sigma	Total	Mean	Sigma	Total	Mean	Sigma	Total	Mean	Sigma	Total	Mean	Sigma
Nevus_img0103a_Combi_T1.png	N	Nevus	1%	103	32	13	2	54464	432	33,622	14153	14	11,314	24	2	0,953	49	3	1,414
nvmelcomp_a_Combi_T1.png	N	Nevus	1%	1197	112	40	1,414	282443	202	32,7	59133	7	6,708	266	2	2,445	258	2	1,863
nvmelcomp_b_Combi_T1.png	N	Nevus	1%	661	78	28	1,423	162505	248	35,88	46893	8	8,775	76	1	0,964	97	1	1,895
nvmelcomp0_Combi_T1	N	Nevus	1%	262	49	16	0,958	46500	212	56,013	61899	16	17,72	43	1	0,816	48	1	1,347
nvmelintra0_Combi_T1.png	N	Nevus	1%	611	80	28	2,121	164810	194	36,988	33313	8	6,481	82	1	1,491	202	3	2,057
nvmelpeq_Combi_T1.png	N	Nevus	1%	484	93	42	5,298	85921	173	23,607	21203	7	6,928	629	7	4,767	275	5	2,257
Naevi_melanocytic3a_Combi_T1.png		Nevus	1%	1309	137	57	4,291	2888352	563	105,748	1912337	20	19,698	2083	8	8,7	2289	12	12,455
Naevi_melanocytic3b_Combi_T1.png		Nevus	1%	1465	162	58	4,963	2257626	548	46,184	1274145	18	18,083	789	5	4,654	1796	8	5,939
Naevi_melanocytic3c_Combi_T1.png		Nevus	1%	212	45	13	476	75792	476	35,485	13053	13	9,644	79	2	1,759	34	1	1,175
no_1thn_Combi_T1.png			1%	4166	413	148	10,944	13952917	477	122,579	45301256	38	40,012	12541	15	12,259	13616	16	13,743
no_2thn_Combi_T1.png			1%	6234	302	120	7,522	14001214	520	138,731	41437398	33	39,85	13252	16	12,435	12034	16	14,092
no_3thn_Combi_T1.png			1%	3378	275	113	8,206	4613709	363	136,26	12402631	29	31,89	6239	12	8,346	6466	14	10,268
no_5thn_Combi_T1.png			1%	4761	257	112	4,588	7782340	463	89,592	18435383	35	33,69	4526	12	9,838	4612	11	10,454

8.5.2 Calculated

File Name	Direction Changes/Number of Edge Points	Total Colour Differences/Total Colours	Colour Min	Colour Max	Edge Points / Lesion Points	Dir. Changes / Edge Points
basal_fig4_Combi_T8.png	0,023885554	0,724213457	199,075	396,925	0,063609656	0,375502008
Basal_img0019a_Combi_T1.png	0,033989749	0,277220329	369,827	458,173	0,082816293	0,410423453
Basalioma_01_Combi_T1.png	0,024135682	1,293693684	262,965	495,035	0,084148728	0,286821705
Basalioma_02_Combi_T1.png	0,030681818	3,699626595	279,183	606,817	0,072727273	0,421875
Basalioma_03_Combi_T4.png	0,034810127	1,554378254	427,963	548,037	0,102848101	0,338461538
Basalioma_04_Combi_T1.png	0,013419659	2,04732895	317,1	606,9	0,038493231	0,348623853
Basalioma_05_Combi_T1.png	0,051172708	3,077745601	212,645	459,355	0,147121535	0,347826087
basalioma_a09f8_Combi_T1.png	0,043652785	1,866982382	170,389	363,611	0,103361766	0,422330097
Carcinoma_basal_005A_Combi_T1.png	0,080555556	1,589725795	351,045	472,955	0,208333333	0,386666667
Carcinoma_basal_02_Combi_T8.png	0,018376367	2,532144948	277,098	566,902	0,053966039	0,340517241
1287melanoma2_Combi_T9.png	0,034258712	4,835898566	320,976	621,024	0,089190786	0,38410596
7melanoma_Combi_T1.png	0,020192809	15,36523911	227,932	540,068	0,056409588	0,357967667
image_a_Combi_T1.png	0,026642984	0,372939526	544,759	691,241	0,062420705	0,426829268
malig2_Combi_T1.png	0,03648863	17,49083148	100,183	439,817	0,112109995	0,325471698
malignant_melanoma_1_Combi_T1.png	0,017526006	5,876950449	236,561	449,439	0,048846676	0,358796296
malignant_melanoma_2_Combi_T6.png	0,018946782	3,564942362	210,288	437,712	0,053078852	0,356955381
Melanoma_04_Combi_T6.png	0,027788209	2,814497279	96,595	267,405	0,07172362	0,387434555
Melanoma_005_Combi_T6.png	0,035629454	8,247102909	277,867	656,133	0,076688157	0,46460177
Melanoma_005a_Combi_T3.png	0,020735156	1,955398371	161,06	356,94	0,063901979	0,324483776
Melanoma_006_Combi_T1.png	0,071022727	9,804183843	274,124	691,876	0,193181818	0,367647059
melanoma_007_Combi_T10.png	0,059961315	3,250116536	304,374	677,626	0,15860735	0,37804878
melanoma_009_Combi_T1.png	0,018436874	5,875817496	316,319	597,681	0,051302605	0,359375
melanoma_01_Combi_T6.png	0,084309133	0,342906036	149,547	224,453	0,180327869	0,467532468
melanoma_012_Combi_T1.png	0,014945919	0,730728683	336,138	521,862	0,059193707	0,252491694
melanoma_014_Combi_T1.png	0,020822397	0,815790809	235,397	364,603	0,049868766	0,41754386
Melanoma_015_Combi_T1.png	0,013131538	1,117791235	339,877	652,123	0,039060761	0,336182336
Melanoma_016_Combi_T1.png	0,023110785	11,1376938	255,097	644,903	0,070799707	0,32642487
Melanoma_017_Combi_T1.png	0,03875969	12,65457079	282,717	595,283	0,101205857	0,382978723
melanoma_018_Combi_T1.png	0,043050431	26,25009817	71,342	374,658	0,097785978	0,440251572
melanoma_019_Combi_T1.png	0,023929896	28,59031896	103,385	616,615	0,065722952	0,364102564
Melanoma_02_Combi_T10.png	0,043478261	3,744396359	140,213	327,787	0,114975845	0,378151261
melanoma_020_Combi_T1.png	0,046788263	12,31597293	237,764	664,236	0,149088025	0,313829787
Melanoma_021_Combi_T3.png	0,024700071	6,355077513	137,508	388,492	0,058103976	0,425101215
melanoma_10_Combi_T1.png	0,033126294	8,238001605	420,705	729,295	0,076190476	0,434782609
melanoma_a09f2_Combi_T1.png	0,025364274	2,459047965	154,449	569,551	0,078251484	0,324137931
melanoma_abdc_01_Combi_T12.png	0,031481481	12,47657038	117,734	352,266	0,078518519	0,400943396
melanoma_abdc_02_Combi_T1.png	0,03048551	42,92062026	57,589	346,411	0,073767407	0,413265306

File Name	Direction Changes/Number of Edge Points	Total Colour Differences/Total Colours	Colour Min	Colour Max	Edge Points / Lesion Points	Dir. Changes / Edge Points
melanoma_abdc_03_Combi_T2.png	0,025411061	11,89666923	81,015	290,985	0,063776781	0,3984375
Melanoma_img0002_Combi_T6.png	0,035582822	29,84523898	101,362	416,638	0,08404908	0,423357664
Melanoma_img0003_Combi_T3.png	0,01582428	2,14864096	150,137	511,863	0,041332074	0,382857143
Melanoma_img0004_Combi_T1.png	0,03373494	2,975875689	137,374	346,626	0,093012048	0,362694301
Melanoma_img0005a_Combi_T1.png	0,030639913	3,682426146	195,874	402,126	0,070498915	0,434615385
Melanoma_img0005b_Combi_T1.png	0,024702448	7,271484384	356,351	711,649	0,056591062	0,436507937
Melanoma_img0010_Combi_T6.png	0,001954844	15,3503896	296,325	553,675	0,088669951	0,022046296
Melanoma_img0013_Combi_T3.png	0,020589775	23,42048539	172,58	545,42	0,049206072	0,418439716
Melanoma_img0014_Combi_T1.png	0,017377172	3,853113872	209,763	456,237	0,041130141	0,422492401
Melanoma_img0019_Combi_T7.png	0,043209877	12,24286152	109,369	424,631	0,098324515	0,439461883
Melanoma_img0020_Combi_T7.png	0,001404654	11,88243533	108,389	351,611	0,060978237	0,023035336
Melanoma_img0021_Combi_T1.png	0,002152838	23,99859383	159,825	502,175	0,079475983	0,027087912
Melanoma_img0024_Combi_T1.png	0,047465438	11,55202662	183,948	580,052	0,105990783	0,447826087
Melanoma_img0030_Combi_T2.png	0,02352577	2,290201835	362,833	725,167	0,064231543	0,36626506
Melanoma_img0033_Combi_T1.png	0,036845984	4,121549893	131,654	380,346	0,106853353	0,344827586
Melanoma_img0048_Combi_T1.png	0,049036778	0,769728822	295,708	422,292	0,111208406	0,440944882
Melanoma_img0054_Combi_T3.png	0,020529017	25,27144017	197,334	434,666	0,053691275	0,382352941
Melanoma_img0056_Combi_T1.png	0,02918208	9,533740824	140,28	481,72	0,073160707	0,398876404
Melanoma_img0056a_Combi_T1.png	0,042867701	14,82597839	194,017	603,983	0,104951959	0,408450704
Melanoma_img0064a_Combi_T3.png	0,036267138	11,81884639	181,051	548,949	0,089341	0,405940594
Melanoma_img0081_Combi_T1.png	0,025084238	1,951378995	116,158	357,842	0,071134407	0,352631579
Melanoma_img0085a_Combi_T1.png	0,016810759	2,341486606	145,43	462,57	0,055235351	0,304347826
Melanoma_img0090_Combi_T5.png	0,027840909	12,45149163	172,328	493,672	0,063068182	0,441441441
Melanoma_img0090a_Combi_T1.png	0,032797428	22,25677021	145,664	472,336	0,10096463	0,324840764
Melanoma_img0092_Combi_T1.png	0,023963731	10,63229194	201,157	578,843	0,064443005	0,371859296
Melanoma_img0095a_Combi_T5.png	0,025269263	5,418158356	179,533	538,467	0,073736537	0,342696629
Melanoma_img0095b_Combi_T5.png	0,028245364	7,602512482	169,279	498,721	0,070470756	0,400809717
Melanoma_img0097_Combi_T1.png	0,022495422	3,379085557	439,062	688,938	0,056500131	0,398148148
Melanoma_img0100_Combi_T1.png	0,019672131	10,4401182	143,663	398,337	0,052459016	0,375
Melanoma_img0100a_Combi_T8.png	0,028642247	11,90726308	197,164	560,836	0,074084274	0,3866171
Melanoma_img0102_Combi_T1.png	0,019182948	12,16405815	112,929	413,071	0,047779751	0,401486989
Melanoma_img0103a_Combi_T1.png	0,034138857	8,02330093	132,314	517,686	0,08208669	0,41588785
Melanoma_img0105_Combi_T1.png	0,020522388	1,088322758	236,012	545,988	0,053871269	0,380952381
melanoma_mal_Combi_T4.png	0,052254098	5,23826492	243,852	420,148	0,12602459	0,414634146
Melanoma_malignant_Combi_T1.png	0,01542192	6,292837055	263,277	436,723	0,042580019	0,362186788
melanoma_nodule_Combi_T1.png	0,016408628	0,085272456	483,565	588,435	0,041153897	0,398713826
melanoma_palpabile_Combi_T1.png	0,043850267	7,155072867	150,276	479,724	0,120855615	0,362831858
Melanoma_XX01A_1_Combi_T1.png	0,018835019	53,40536404	25,379	368,621	0,048133938	0,391304348
Melanoma_XX04_Combi_T1.png	0,024009604	63,42676192	49,471	308,529	0,058343337	0,411522634
melanoma-1_Combi_T4.png	0,032209361	2,385931101	283,753	390,247	0,071716155	0,449122807
melanoma10_Combi_T9.png	0,016127343	1,455370583	223,701	404,299	0,039480574	0,408488064
Melanoma1707_Combi_T6.png	0,013766234	5,763108242	188,173	647,827	0,038441558	0,358108108
melanoma-2_Combi_T10.png	0,03198594	0,259374723	210,195	299,805	0,089279438	0,358267717
melanoma3_Combi_T1.png	0,024972341	1,332716465	234,007	365,993	0,064959697	0,384428224
Melanoma3200_Combi_T9.png	0,031178707	0,968748419	347,627	582,373	0,079847909	0,39047619

File Name	Direction Changes/Number of Edge Points	Total Colour Differences/Total Colours	Colour Min	Colour Max	Edge Points / Lesion Points	Dir. Changes / Edge Points
melanoma4_Combi_T6.png	0,025935374	0,510121473	172,259	293,741	0,066964286	0,387301587
melanoma8_Combi_T10.png	0,036987223	1,796781275	227,636	434,364	0,098856759	0,37414966
melanoma-fig1_Combi_T2.pn	0,024382991	7,125967622	272,625	561,375	0,058876004	0,414141414
melanoma-fig2a_Combi_T1.pn	0,025938567	4,832863763	152,776	349,224	0,060068259	0,431818182
melanoma-fig3_Combi_T1.pn	0,036016007	2,152307824	154,745	467,255	0,093819475	0,383886256
melanoma-fig4_Combi_T3.pn	0,035714286	0,845979515	468,398	673,602	0,094684385	0,377192982
n138a_Combi_T1.png	0,020710059	2,376789814	148,697	307,303	0,044167371	0,468899522
Escamosas_img0046_Combi_T1.png	0,001510813	2,974413298	143,618	300,382	0,074394464	0,418604651
Atypical_mole_001_Combi_T6.png	0,066914498	1,972067966	205,344	338,656	0,200743494	0,333333333
Atypical_mole_002_Combi_T4.png	0,043977055	1,527872451	227,79	400,21	0,100382409	0,438095238
keratosis1_Combi_T1.png	0,03407155	11,23912937	192,146	465,854	0,095400341	0,357142857
Queratose_img0088_Combi_T4.png	0,021648045	1,861465951	149,085	350,915	0,062849162	0,344444444
Queratose_img0093_Combi_T1.png	0,014431538	2,338213849	224,871	351,129	0,045054558	0,3203125
Queratose_img0094_Combi_T1.png	0,031618887	1,315725012	336,753	589,247	0,078836425	0,401069519
Queratose_img0095_Combi_T12.png	0,01510534	2,347959085	167,011	434,989	0,047436067	0,318435754
seborrheic_keratosis_02_Combi_T1.png	0,016073871	1,558131507	221,859	446,141	0,046340629	0,346863469
seborrheic_keratosis_03_Combi_T4.png	0,017552658	1,995265626	136,044	253,956	0,044466734	0,394736842
Atypical nevus_01_Combi_T1.png	0,037735849	6,763953729	303,104	656,896	0,161425577	0,233766234
Benign nevus_01_Combi_T4.png	0,027101516	1,162362868	250,944	367,056	0,06361966	0,42599278
Benign nevus_02_Combi_T1.png	0,051666667	1,930547193	145,975	258,025	0,128333333	0,402597403
Benign nevus_03_Combi_T11.png	0,035335689	1,054085718	227,947	478,053	0,102473498	0,344827586
nevi4_Combi_T1.png	0,062927497	4,727162619	223,153	400,847	0,135430917	0,464646465
nevi4a_Combi_T1.png	0,037367994	7,761180018	134,896	301,104	0,091795288	0,407079646
nevo_03_Combi_T3.png	0,076190476	9,753139389	127,479	394,521	0,20952381	0,363636364
nevo_04_Combi_T1.png	0,029411765	2,513470964	69,122	214,878	0,089009288	0,330434783
nevo_congenito_Combi_T1.png	0,077868852	0,396946438	296,085	363,915	0,213114754	0,365384615
nevo_img0030_Combi_T1.png	0,023854363	0,945290337	247,752	346,248	0,079723792	0,299212598
Nevo_img0031_Combi_T10.png	0,020828042	1,506389255	168,804	503,196	0,055626111	0,374429224
nevo_img0084_Combi_T2.png	0,023566655	27,70947688	101,695	404,305	0,064016884	0,368131868
nevo_lentigginoso_Combi_T1.png	0,073170732	0,701818283	485,154	686,846	0,18699187	0,391304348
nevoa3_small_Combi_T1.png	0,031313819	1,586957178	125,29	400,71	0,085091899	0,368
nevoc2_small_Combi_T1.png	0,040141677	0,822674427	310,413	465,587	0,108618654	0,369565217
nevodis_small_Combi_T0.png	0,035667107	0,494632575	300,056	457,944	0,122853369	0,290322581
nevosp1_small_Combi_T1.png	0,050251256	0,236117504	370,945	421,055	0,120603015	0,416666667
nevosu_small1_Combi_T3.png	0,037278107	0,180338577	457,888	562,112	0,115384615	0,323076923
nevou5_small_Combi_T1.png	0,037257824	0,523794191	233,142	370,858	0,12071535	0,308641975
Nevus_003_Combi_T1.png	0,044776119	0,762383473	363,373	472,627	0,253731343	0,176470588
Nevus_img0085_Combi_T4.png	0,105960265	0,593035172	426,198	503,802	0,231788079	0,457142857
Nevus_img0085b_Combi_T1.png	0,065420561	0,489406612	411,933	500,067	0,200934579	0,325581395
Nevus_img0103_Combi_T4.png	0,125	0,490169349	356,242	421,758	0,35	0,357142857
Nevus_img0103a_Combi_T1.png	0,126213592	0,259859724	398,378	465,622	0,310679612	0,40625
nvmelcomp_a_Combi_T1.png	0,033416876	0,209362597	169,3	234,7	0,093567251	0,357142857
nvmelcomp_b_Combi_T1.png	0,042360061	0,288563429	212,12	283,88	0,118003026	0,358974359
nvmelcomp0_Combi_T1	0,061068702	1,33116129	155,987	268,013	0,187022901	0,326530612
nvmelintra0_Combi_T1.png	0,045826514	0,202129725	157,012	230,988	0,130932897	0,35

File Name	Direction Changes/Number of Edge Points	Total Colour Differences/Total Colours	Colour Min	Colour Max	Edge Points / Lesion Points	Dir. Changes / Edge Points
nvmelpeq_Combi_T1.png	0,08677686	0,246773199	149,393	196,607	0,19214876	0,451612903
Naevi_melanocytic3a_Combi_T1.png	0,043544691	0,662085854	457,252	668,748	0,104660046	0,416058394
Naevi_melanocytic3b_Combi_T1.png	0,039590444	0,564373816	501,816	594,184	0,110580205	0,358024691
Naevi_melanocytic3c_Combi_T1.png	0,061320755	0,172221343	440,515	511,485	0,212264151	0,288888889
no_1thn_Combi_T1.png	0,035525684	3,246722961	354,421	599,579	0,099135862	0,358353511
no_2thn_Combi_T1.png	0,019249278	2,959557507	381,269	658,731	0,048444017	0,397350993
no_3thn_Combi_T1.png	0,033451747	2,688212672	226,74	499,26	0,081409118	0,410909091
no_5thn_Combi_T1.png	0,02352447	2,368874015	373,408	552,592	0,053980256	0,435797665

8.5.3 Evaluation

Evaluator: weka.attributeSelection.GainRatioAttributeEval

GainR(Class, Attribute) = (H(Class) - H(Class | Attribute)) / H(Attribute)

Search: weka.attributeSelection.Ranker -T -1.7976931348623157E308 -N -1

Relation: Image_Features-weka.filters.unsupervised.attribute.Remove-R1

Instances: 136

Attributes: 22

Malignant
Lesion_Nr_Points
Edge_Nr_Points
Dir_Changes
Rays_Sigma
C_Total
C_Mean
C_Sigma
CD_Total
CD_Mean
CD_Sigma
X_Total
X_Mean
X_Sigma
Y_Total
Y_Mean
Y_Sigma
DC_EP
TCD_TC
Cmin
Cmax
EP_LP

Evaluation mode: evaluate on all training data

=== Attribute Selection on all input data ===

Search Method:

Attribute ranking.

Attribute Evaluator (supervised, Class (nominal): 1 Malignant):

Gain Ratio feature evaluator

Ranked attributes:

0.536 17 Y_Sigma
0.53 15 Y_Total
0.424 3 Edge_Nr_Points
0.38 4 Dir_Changes
0.374 5 Rays_Sigma
0.368 14 X_Sigma
0.318 2 Lesion_Nr_Points
0.307 9 CD_Total
0.301 12 X_Total
0.283 6 C_Total
0.276 16 Y_Mean
0.274 13 X_Mean
0.236 22 EP_LP
0.231 10 CD_Mean
0.229 11 CD_Sigma
0.22 8 C_Sigma
0.212 19 TCD_TC
0 21 Cmax
0 18 DC_EP
0 20 Cmin
0 7 C_Mean

Selected attributes: 17,15,3,4,5,14,2,9,12,6,16,13,22,10,11,8,19,21,18,20,7 : 21

Evaluator: weka.attributeSelection.GainRatioAttributeEval

GainR(Class, Attribute) = (H(Class) - H(Class | Attribute)) / H(Attribute)

Search: weka.attributeSelection.Ranker -T -1.7976931348623157E308 -N -1

Relation: Image_Features-weka.filters.unsupervised.attribute.Remove-R1

Instances: 136

Attributes: 22

Malignant
Lesion_Nr_Points
Edge_Nr_Points
Dir_Changes
Rays_Sigma
C_Total
C_Mean
C_Sigma
CD_Total
CD_Mean
CD_Sigma
X_Total
X_Mean
X_Sigma
Y_Total
Y_Mean
Y_Sigma
DC_EP
TCD_TC
Cmin
Cmax
EP_LP

Evaluation mode: 10-fold cross-validation

=== Attribute selection 10 fold cross-validation (stratified), seed: 1 ===

average merit	average rank	attribute
0.534 +- 0.02	1.4 +- 0.66	17 Y_Sigma
0.532 +- 0.027	1.7 +- 0.46	15 Y_Total
0.422 +- 0.029	3.8 +- 1.17	3 Edge_Nr_Points
0.379 +- 0.033	5.2 +- 1.25	5 Rays_Sigma
0.381 +- 0.035	5.3 +- 0.64	4 Dir_Changes
0.374 +- 0.057	5.9 +- 2.21	14 X_Sigma
0.344 +- 0.083	7.7 +- 3.26	6 C_Total
0.329 +- 0.04	8 +- 1.67	12 X_Total
0.33 +- 0.041	8 +- 1.67	2 Lesion_Nr_Points
0.312 +- 0.017	8.6 +- 1.28	9 CD_Total
0.276 +- 0.02	12.1 +- 1.92	13 X_Mean
0.267 +- 0.018	12.1 +- 2.17	16 Y_Mean
0.246 +- 0.025	13.6 +- 1.74	22 EP_LP
0.246 +- 0.023	14 +- 1.26	11 CD_Sigma
0.238 +- 0.023	14.9 +- 1.45	8 C_Sigma
0.233 +- 0.018	15 +- 1.26	10 CD_Mean
0.214 +- 0.017	16.4 +- 1.28	19 TCD_TC
0.055 +- 0.1	17.8 +- 2.4	21 Cmax
0 +- 0	19.4 +- 0.66	7 C_Mean
0 +- 0	19.6 +- 1.28	18 DC_EP
0 +- 0	20.5 +- 0.5	20 Cmin

Evaluator: weka.attributeSelection.InfoGainAttributeEval

InfoGain(Class,Attribute) = H(Class) - H(Class | Attribute)

Search: weka.attributeSelection.Ranker -T -1.7976931348623157E308 -N -1

Relation: Image_Features-weka.filters.unsupervised.attribute.Remove-R1

Instances: 136

Attributes: 22

Malignant
Lesion_Nr_Points
Edge_Nr_Points
Dir_Changes
Rays_Sigma
C_Total
C_Mean
C_Sigma
CD_Total
CD_Mean
CD_Sigma
X_Total
X_Mean
X_Sigma
Y_Total
Y_Mean
Y_Sigma
DC_EP
TCD_TC
Cmin
Cmax
EP_LP

Evaluation mode: evaluate on all training data

=== Attribute Selection on all input data ===

Search Method:

Attribute ranking.

Attribute Evaluator (supervised, Class (nominal): 1 Malignant):

Information Gain Ranking Filter

Ranked attributes:

0.453 9 CD_Total
0.427 6 C_Total
0.403 15 Y_Total
0.369 17 Y_Sigma
0.362 16 Y_Mean
0.328 4 Dir_Changes
0.328 3 Edge_Nr_Points
0.307 14 X_Sigma
0.288 2 Lesion_Nr_Points
0.275 12 X_Total
0.274 5 Rays_Sigma
0.259 13 X_Mean
0.231 10 CD_Mean
0.228 11 CD_Sigma
0.208 19 TCD_TC
0.2 8 C_Sigma
0.194 22 EP_LP
0 21 Cmax
0 18 DC_EP
0 20 Cmin
0 7 C_Mean

Selected attributes: 9,6,15,17,16,4,3,14,2,12,5,13,10,11,19,8,22,21,18,20,7 : 21

Evaluator: weka.attributeSelection.InfoGainAttributeEval

InfoGain(Class,Attribute) = H(Class) - H(Class | Attribute)

Search: weka.attributeSelection.Ranker -T -1.7976931348623157E308 -N -1
Relation: Image_Features-weka.filters.unsupervised.attribute.Remove-R1
Instances: 136
Attributes: 22

- Malignant
- Lesion_Nr_Points
- Edge_Nr_Points
- Dir_Changes
- Rays_Sigma
- C_Total
- C_Mean
- C_Sigma
- CD_Total
- CD_Mean
- CD_Sigma
- X_Total
- X_Mean
- X_Sigma
- Y_Total
- Y_Mean
- Y_Sigma
- DC_EP
- TCD_TC
- Cmin
- Cmax
- EP_LP

Evaluation mode: 10-fold cross-validation

=== Attribute selection 10 fold cross-validation (stratified), seed: 1 ===

average merit	average rank	attribute
0.459 +- 0.024	1.1 +- 0.3	9 CD_Total
0.408 +- 0.019	2.7 +- 0.64	15 Y_Total
0.414 +- 0.04	3.2 +- 1.89	6 C_Total
0.371 +- 0.021	4.2 +- 0.75	17 Y_Sigma
0.367 +- 0.018	4.9 +- 0.83	16 Y_Mean
0.333 +- 0.026	6.6 +- 1.43	3 Edge_Nr_Points
0.33 +- 0.028	6.6 +- 1.02	4 Dir_Changes
0.332 +- 0.032	6.8 +- 1.89	14 X_Sigma
0.292 +- 0.028	9.5 +- 0.92	2 Lesion_Nr_Points
0.282 +- 0.016	10.4 +- 0.92	12 X_Total
0.279 +- 0.024	10.8 +- 1.33	5 Rays_Sigma
0.262 +- 0.015	12.2 +- 0.98	13 X_Mean
0.236 +- 0.015	13 +- 0.89	11 CD_Sigma
0.232 +- 0.018	13.1 +- 1.58	10 CD_Mean
0.208 +- 0.013	15.7 +- 0.78	8 C_Sigma
0.21 +- 0.017	15.8 +- 0.87	19 TCD_TC
0.199 +- 0.02	16.4 +- 0.8	22 EP_LP
0.032 +- 0.049	18 +- 0	21 Cmax
0 +- 0	20 +- 0	20 Cmin
0 +- 0	20 +- 1	7 C_Mean
0 +- 0	20 +- 1	18 DC_EP

Evaluator: weka.attributeSelection.PrincipalComponents -R 0.95 -A 5

Search: weka.attributeSelection.Ranker -T -1.7976931348623157E308 -N -1

Relation: Image_Features-weka.filters.unsupervised.attribute.Remove-R1

Instances: 136

Attributes: 22

- Malignant
- Lesion_Nr_Points
- Edge_Nr_Points
- Dir_Changes
- Rays_Sigma
- C_Total
- C_Mean
- C_Sigma
- CD_Total
- CD_Mean
- CD_Sigma
- X_Total
- X_Mean
- X_Sigma
- Y_Total
- Y_Mean
- Y_Sigma
- DC_EP
- TCD_TC
- Cmin
- Cmax
- EP_LP

Evaluation mode: evaluate on all training data

=== Attribute Selection on all input data ===

Search Method:

Attribute ranking.

Attribute Evaluator (unsupervised):

Principal Components Attribute Transformer

Correlation matrix

1 0.93 0.9 0.71 0.21 0.02 0.23 0.15 0.09 0.11 0.18 0.68 0.71 0.21 0.68 0.73 0.05 0.07 -0.09 0.11 -0.7

0.93 1 0.98 0.78 0.16 0.02 0.27 0.15 0.16 0.17 0.17 0.72 0.75 0.16 0.73 0.76 0.08 0.09 -0.11 0.13 -0.73

0.9 0.98 1 0.75 0.14 0 0.25 0.15 0.19 0.2 0.15 0.69 0.71 0.14 0.72 0.76 0.26 0.12 -0.12 0.11 -0.71

0.71 0.78 0.75 1 0.13 0.03 0.24 0.16 0.12 0.14 0.17 0.77 0.75 0.14 0.71 0.67 0.01 0.05 -0.09 0.13 -0.46

0.21 0.16 0.14 0.13 1 0.29 0.16 0.61 -0.05 0.01 0.85 0.3 0.27 0.99 0.39 0.34 -0.03 -0.06 0.2 0.31 -0.15

0.02 0.02 0 0.03 0.29 1 0.19 0.32 -0.11 -0.02 0.28 0.29 0.28 0.28 0.36 0.35 -0.07 -0.25 0.88 0.91 0.06

0.23 0.27 0.25 0.24 0.16 0.19 1 0.31 0.67 0.74 0.19 0.45 0.48 0.17 0.42 0.45 -0.03 0.46 -0.3 0.57 -0.38

0.15 0.15 0.15 0.16 0.61 0.32 0.31 1 0.16 0.26 0.7 0.32 0.28 0.63 0.33 0.28 0.05 0.08 0.16 0.39 -0.16

0.09 0.16 0.19 0.12 -0.05 -0.11 0.67 0.16 1 0.97 -0.04 0.09 0.12 -0.06 0.1 0.12 0.17 0.86 -0.43 0.18 -0.26

0.11 0.17 0.2 0.14 0.01 -0.02 0.74 0.26 0.97 1 0.04 0.16 0.18 0.01 0.15 0.16 0.14 0.86 -0.37 0.29 -0.26

0.18 0.17 0.15 0.17 0.85 0.28 0.19 0.7 -0.04 0.04 1 0.42 0.37 0.9 0.39 0.33 0.02 -0.05 0.18 0.32 -0.15

0.68 0.72 0.69 0.77 0.3 0.29 0.45 0.32 0.09 0.16 0.42 1 0.96 0.33 0.86 0.82 -0.01 -0.04 0.07 0.43 -0.55

0.71 0.75 0.71 0.75 0.27 0.28 0.48 0.28 0.12 0.18 0.37 0.96 1 0.29 0.83 0.83 -0.01 -0.02 0.04 0.43 -0.59

0.21 0.16 0.14 0.14 0.99 0.28 0.17 0.63 -0.06 0.01 0.9 0.33 0.29 1 0.4 0.35 -0.03 -0.06 0.19 0.31 -0.15

0.68 0.73 0.72 0.71 0.39 0.36 0.42 0.33 0.1 0.15 0.39 0.86 0.83 0.4 1 0.95 0.09 -0.03 0.14 0.48 -0.58

0.73 0.76 0.76 0.67 0.34 0.35 0.45 0.28 0.12 0.16 0.33 0.82 0.83 0.35 0.95 1 0.11 -0.03 0.12 0.48 -0.63

0.05 0.08 0.26 0.01 -0.03 -0.07 -0.03 0.05 0.17 0.14 0.02 -0.01 -0.01 -0.03 0.09 0.11 1 0.14 -0.06 -0.07 -0.04

0.07 0.09 0.12 0.05 -0.06 -0.25 0.46 0.08 0.86 0.86 -0.05 -0.04 -0.02 -0.06 -0.03 -0.03 0.14 1 -0.46 -0.01 -0.2

-0.09 -0.11 -0.12 -0.09 0.2 0.88 -0.3 0.16 -0.43 -0.37 0.18 0.07 0.04 0.19 0.14 0.12 -0.06 -0.46 1 0.61 0.24

0.11 0.13 0.11 0.13 0.31 0.91 0.57 0.39 0.18 0.29 0.32 0.43 0.43 0.31 0.48 0.48 -0.07 -0.01 0.61 1 -0.11

-0.7 -0.73 -0.71 -0.46 -0.15 0.06 -0.38 -0.16 -0.26 -0.26 -0.15 -0.55 -0.59 -0.15 -0.58 -0.63 -0.04 -0.2 0.24 -0.11 1

Eigenvalue	proportion	cumulative	
8.08447	0.38497	0.38497	-0.32Y_Mean-0.32Y_Sigma-0.315X_Mean-0.315X_Sigma-0.296Edge_Nr_Points...
3.93587	0.18742	0.5724	-0.399Cmin+0.344TCD_TC+0.336CD_Mean-0.332C_Mean+0.298CD_Sigma...
3.08513	0.14691	0.71931	0.397CD_Sigma+0.365CD_Mean+0.33 TCD_TC+0.317C_Sigma+0.282CD_Total...
2.03327	0.09682	0.81613	0.428C_Mean+0.427Cmax-0.376Y_Total-0.368C_Total-0.348X_Total...
1.09435	0.05211	0.86824	0.887DC_EP-0.229C_Sigma+0.213Dir_Changes+0.206Cmin-0.137X_Sigma...
0.67487	0.03214	0.90038	-0.339EP_LP+0.336Lesion_Nr_Points-0.328DC_EP-0.324X_Mean+0.303Cmin...
0.57465	0.02736	0.92774	-0.555Rays_Sigma-0.542EP_LP+0.327C_Sigma-0.232TCD_TC-0.214CD_Total...
0.45373	0.02161	0.94935	0.805CD_Total-0.311C_Total-0.281Y_Total-0.251EP_LP-0.217TCD_TC...
0.27284	0.01299	0.96234	0.472EP_LP+0.35 C_Sigma-0.34X_Mean-0.322X_Sigma-0.303TCD_TC...

Eigenvectors

V1	V2	V3	V4	V5	V6	V7	V8	V9	
-0.282	0.0951	-0.2163	-0.0676	0.0233	0.3359	-0.0294	0.0093	0.2498	Lesion_Nr_Points
-0.296	0.1235	-0.2133	-0.0341	0.0461	0.2608	-0.0852	0.038	0.1845	Edge_Nr_Points
-0.2889	0.1405	-0.2041	-0.0396	0.2127	0.1967	-0.0517	0.0457	0.1769	Dir_Changes
-0.2651	0.0859	-0.1901	-0.0024	-0.0722	-0.1731	-0.555	-0.0765	0.0691	Rays_Sigma
-0.1635	-0.2737	0.2059	-0.368	-0.0107	0.12	0.0696	-0.3114	0.149	C_Total
-0.1114	-0.3325	0.1368	0.428	0.0991	0.1733	-0.0291	-0.0205	-0.0297	C_Mean
-0.1935	0.1588	0.3171	0.1671	-0.2292	-0.28	0.3271	0.0468	0.3496	C_Sigma
-0.1607	-0.1733	0.2824	-0.1954	0.0654	0.0003	-0.2139	0.8049	0.1429	CD_Total
-0.0922	0.336	0.3651	0.0842	0.0574	0.0844	-0.1037	-0.088	-0.0285	CD_Mean
-0.1151	0.2982	0.3971	0.1034	0.0338	0.0638	-0.1346	-0.0192	0.0002	CD_Sigma
-0.1737	-0.2621	0.2124	-0.348	0.006	-0.0921	-0.0603	0.004	-0.2915	X_Total
-0.3153	-0.0291	-0.0879	0.0647	-0.1289	-0.3243	-0.1232	0.015	-0.3402	X_Mean
-0.3153	-0.0038	-0.0945	0.0803	-0.1369	-0.2863	-0.0466	0.0121	-0.322	X_Sigma
-0.1672	-0.2767	0.21	-0.3764	-0.0148	0.0758	0.0604	-0.2807	0.071	Y_Total
-0.3201	-0.0526	-0.0728	0.068	0.0281	-0.1608	0.0822	-0.1483	0.0308	Y_Mean
-0.3196	-0.0281	-0.0866	0.0932	0.0457	-0.0957	0.2135	-0.1491	0.1466	Y_Sigma
-0.0283	0.0887	0.0174	-0.0555	0.8867	-0.3281	0.1508	-0.0063	-0.0318	DC_EP
-0.0458	0.3439	0.3302	-0.0228	0.0685	0.2545	-0.2323	-0.2168	-0.3034	TCD_TC

-0.0151 -0.3992 -0.0198 0.3352 0.2064 0.303 -0.1856 -0.0424 -0.1971 Cmin
-0.1734 -0.2122 0.2458 0.4271 -0.0121 0.0289 0.1113 0.0022 0.1201 Cmax
0.2453 -0.1608 0.0934 0.0706 0.0602 -0.3391 -0.5416 -0.2506 0.4719 EP_LP

Ranked attributes:

0.615 1 -0.32Y_Mean-0.32Y_Sigma-0.315X_Mean-0.315X_Sigma-0.296Edge_Nr_Points...
0.4276 2 -0.399Cmin+0.344TCD_TC+0.336CD_Mean-0.332C_Mean+0.298CD_Sigma...
0.2807 3 0.397CD_Sigma+0.365CD_Mean+0.33 TCD_TC+0.317C_Sigma+0.282CD_Total...
0.1839 4 0.428C_Mean+0.427Cmax-0.376Y_Total-0.368C_Total-0.348X_Total...
0.1318 5 0.887DC_EP-0.229C_Sigma+0.213Dir_Changes+0.206Cmin-0.137X_Sigma...
0.0996 6 -0.339EP_LP+0.336Lesion_Nr_Points-0.328DC_EP-0.324X_Mean+0.303Cmin...
0.0723 7 -0.555Rays_Sigma-0.542EP_LP+0.327C_Sigma-0.232TCD_TC-0.214CD_Total...
0.0507 8 0.805CD_Total-0.311C_Total-0.281Y_Total-0.251EP_LP-0.217TCD_TC...
0.0377 9 0.472EP_LP+0.35 C_Sigma-0.34X_Mean-0.322X_Sigma-0.303TCD_TC...

Selected attributes: 1,2,3,4,5,6,7,8,9 : 9

8.6 Appendix F – Weka© result reports

```
Scheme:       weka.classifiers.bayes.NaiveBayes
Relation:     Image_Features
Instances:    136
Attributes:   23
              File_Name
              Malignant
              Lesion_Nr_Points
              Edge_Nr_Points
              Dir_Changes
              Rays_Sigma
              C_Total
              C_Mean
              C_Sigma
              CD_Total
              CD_Mean
              CD_Sigma
              X_Total
              X_Mean
              X_Sigma
              Y_Total
              Y_Mean
              Y_Sigma
              DC_EP
              TCD_TC
              Cmin
              Cmax
              EP_LP
Test mode:    split 50% train, remainder test
```

```
=== Classifier model (full training set) ===

Naive Bayes Classifier

Class Y: Prior probability = 0.67

File_Name:  Discrete Estimator. Counts =  2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1  (Total = 228)
Lesion_Nr_Points:  Normal Distribution. Mean = 4025.8509 StandardDev =
2562.2427 WeightSum = 92 Precision = 85.59701492537313
Edge_Nr_Points:  Normal Distribution. Mean = 250.3705 StandardDev = 97.2539
WeightSum = 92 Precision = 4.086956521739131
Dir_Changes:  Normal Distribution. Mean = 96.9324 StandardDev = 38.5762
WeightSum = 92 Precision = 2.2222222222222223
Rays_Sigma:  Normal Distribution. Mean = 6.3679 StandardDev = 3.5685
WeightSum = 92 Precision = 0.1331777777777778
C_Total:  Normal Distribution. Mean = 34152562.3894 StandardDev =
117105750.3848 WeightSum = 92 Precision = 6966819.822222223
C_Mean:  Normal Distribution. Mean = 355.337 StandardDev = 103.7689
WeightSum = 92 Precision = 4.25
C_Sigma:  Normal Distribution. Mean = 134.2329 StandardDev = 46.5182
WeightSum = 92 Precision = 1.7259851851851853
CD_Total:  Normal Distribution. Mean = 143567047.6 StandardDev =
365555627.7094 WeightSum = 92 Precision = 1.51123208E7
CD_Mean:  Normal Distribution. Mean = 45.1863 StandardDev = 26.0642
WeightSum = 92 Precision = 1.7142857142857142
CD_Sigma:  Normal Distribution. Mean = 46.4196 StandardDev = 25.1631
WeightSum = 92 Precision = 0.7848923076923077
X_Total:  Normal Distribution. Mean = 37618.1262 StandardDev = 110785.8261
WeightSum = 92 Precision = 5243.738805970149
X_Mean:  Normal Distribution. Mean = 11.952 StandardDev = 5.4169 WeightSum
= 92 Precision = 1.2083333333333333
X_Sigma:  Normal Distribution. Mean = 9.606 StandardDev = 4.0019 WeightSum
```

= 92 Precision = 0.16518796992481205
Y_Total: Normal Distribution. Mean = 33868.653 StandardDev = 111972.5405 WeightSum = 92 Precision = 6424.569230769231
Y_Mean: Normal Distribution. Mean = 10.9447 StandardDev = 4.7645 WeightSum = 92 Precision = 1.0909090909090908
Y_Sigma: Normal Distribution. Mean = 8.7233 StandardDev = 3.5869 WeightSum = 92 Precision = 0.12616666666666665
DC_EP: Normal Distribution. Mean = 0.388 StandardDev = 0.041 WeightSum = 92 Precision = 0.0018078015343511453
TCD_TC: Normal Distribution. Mean = 8.5271 StandardDev = 10.8196 WeightSum = 92 Precision = 0.4691962182518518
Cmin: Normal Distribution. Mean = 221.0083 StandardDev = 103.8611 WeightSum = 92 Precision = 3.847259259259259
Cmax: Normal Distribution. Mean = 489.4982 StandardDev = 122.7524 WeightSum = 92 Precision = 3.945837037037036
EP_LP: Normal Distribution. Mean = 0.0765 StandardDev = 0.0294 WeightSum = 92 Precision = 0.002307840311111111

Class N: Prior probability = 0.33

File_Name: Discrete Estimator. Counts = 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 (Total = 180)
Lesion_Nr_Points: Normal Distribution. Mean = 1496.0024 StandardDev = 1792.7591 WeightSum = 44 Precision = 85.59701492537313
Edge_Nr_Points: Normal Distribution. Mean = 116.8498 StandardDev = 77.5412 WeightSum = 44 Precision = 4.086956521739131
Dir_Changes: Normal Distribution. Mean = 42.2727 StandardDev = 28.0551 WeightSum = 44 Precision = 2.2222222222222223
Rays_Sigma: Normal Distribution. Mean = 2.7392 StandardDev = 1.6928 WeightSum = 44 Precision = 0.1331777777777778
C_Total: Normal Distribution. Mean = 316673.6283 StandardDev = 1451180.8722 WeightSum = 44 Precision = 6966819.822222223

C_Mean: Normal Distribution. Mean = 337.6818 StandardDev = 109.3215 WeightSum = 44 Precision = 4.25
C_Sigma: Normal Distribution. Mean = 75.5511 StandardDev = 41.0738 WeightSum = 44 Precision = 1.7259851851851853
CD_Total: Normal Distribution. Mean = 1030385.5091 StandardDev = 4989071.3536 WeightSum = 44 Precision = 1.51123208E7
CD_Mean: Normal Distribution. Mean = 22.8701 StandardDev = 14.2912 WeightSum = 44 Precision = 1.7142857142857142
CD_Sigma: Normal Distribution. Mean = 22.5657 StandardDev = 15.9931 WeightSum = 44 Precision = 0.7848923076923077
X_Total: Normal Distribution. Mean = 2741.0453 StandardDev = 11804.4298 WeightSum = 44 Precision = 5243.738805970149
X_Mean: Normal Distribution. Mean = 5.108 StandardDev = 3.9312 WeightSum = 44 Precision = 1.2083333333333333
X_Sigma: Normal Distribution. Mean = 3.9683 StandardDev = 3.1003 WeightSum = 44 Precision = 0.16518796992481205
Y_Total: Normal Distribution. Mean = 1022.0906 StandardDev = 4096.1769 WeightSum = 44 Precision = 6424.569230769231
Y_Mean: Normal Distribution. Mean = 4.5124 StandardDev = 3.4229 WeightSum = 44 Precision = 1.0909090909090908
Y_Sigma: Normal Distribution. Mean = 3.7793 StandardDev = 3.0085 WeightSum = 44 Precision = 0.12616666666666665
DC_EP: Normal Distribution. Mean = 0.3669 StandardDev = 0.0528 WeightSum = 44 Precision = 0.0018078015343511453
TCD_TC: Normal Distribution. Mean = 2.3566 StandardDev = 4.5868 WeightSum = 44 Precision = 0.4691962182518518
Cmin: Normal Distribution. Mean = 262.1383 StandardDev = 114.882 WeightSum = 44 Precision = 3.847259259259259
Cmax: Normal Distribution. Mean = 413.1471 StandardDev = 119.3533 WeightSum = 44 Precision = 3.945837037037036
EP_LP: Normal Distribution. Mean = 0.1351 StandardDev = 0.0698 WeightSum = 44 Precision = 0.002307840311111111

Time taken to build model: 0.02 seconds

```
=== Predictions on test split ===

 inst#,     actual, predicted, error, probability distribution
      1        1:Y        1:Y          *1      0
      2        2:N        2:N           0     *1
      3        1:Y        2:N      +    0     *1
      4        1:Y        2:N      +    0     *1
      5        2:N        2:N           0     *1
      6        2:N        2:N           0     *1
      7        2:N        2:N           0.005 *0.995
      8        1:Y        1:Y          *1      0
      9        1:Y        2:N      +    0     *1
     10        1:Y        2:N      +    0     *1
     11        1:Y        2:N      +    0     *1
     12        1:Y        2:N      +    0     *1
     13        1:Y        1:Y          *0.999  0.001
     14        1:Y        1:Y          *1      0
     15        1:Y        1:Y          *1      0
     16        2:N        2:N           0     *1
     17        1:Y        1:Y          *1      0
     18        2:N        2:N           0     *1
     19        1:Y        1:Y          *1      0
     20        1:Y        1:Y          *0.949  0.051
     21        1:Y        1:Y          *1      0
     22        2:N        2:N           0     *1
     23        1:Y        2:N      +    0     *1
     24        1:Y        1:Y          *1      0
     25        1:Y        1:Y          *0.753  0.247
     26        1:Y        1:Y          *1      0
     27        2:N        2:N           0     *1
     28        1:Y        1:Y          *1      0
     29        1:Y        2:N      +    0     *1
     30        2:N        2:N           0     *1
     31        2:N        2:N           0     *1
     32        1:Y        1:Y          *0.828  0.172
     33        2:N        2:N           0     *1
```

```
34       2:N       2:N          0     *1
35       1:Y       1:Y         *1      0
36       2:N       2:N          0     *1
37       1:Y       2:N      +   0     *1
38       1:Y       2:N      +   0.301 *0.699
39       1:Y       2:N      +   0     *1
40       2:N       2:N          0     *1
41       1:Y       2:N      +   0     *1
42       2:N       2:N          0     *1
43       1:Y       1:Y         *1      0
44       1:Y       1:Y         *1      0
45       2:N       2:N          0     *1
46       2:N       2:N          0     *1
47       2:N       2:N          0     *1
48       2:N       2:N          0     *1
49       1:Y       1:Y         *1      0
50       1:Y       2:N      +   0.008 *0.992
51       1:Y       1:Y         *1      0
52       2:N       2:N          0     *1
53       2:N       2:N          0     *1
54       1:Y       1:Y         *1      0
55       1:Y       1:Y         *1      0
56       1:Y       1:Y         *1      0
57       2:N       2:N          0     *1
58       2:N       2:N          0     *1
59       1:Y       1:Y         *1      0
60       1:Y       1:Y         *1      0
61       1:Y       1:Y         *1      0
62       1:Y       1:Y         *1      0
63       1:Y       2:N      +   0.009 *0.991
64       1:Y       1:Y         *0.723  0.277
65       1:Y       1:Y         *1      0
66       1:Y       1:Y         *1      0
67       2:N       2:N          0.017 *0.983
68       1:Y       1:Y         *1      0
```

```
=== Evaluation on test split ===
=== Summary ===

Correctly Classified Instances          54               79.4118 %
Incorrectly Classified Instances        14               20.5882 %
Kappa statistic                          0.602
K&B Relative Info Score               3641.2273 %
K&B Information Score                   32.0898 bits        0.4719
bits/instance
Class complexity | order 0              64.3284 bits         0.946
bits/instance
Class complexity | scheme              307.3059 bits         4.5192
bits/instance
Complexity improvement        (Sf)     -242.9775 bits        -3.5732
bits/instance
Mean absolute error                      0.2125
Root mean squared error                  0.4476
Relative absolute error                 48.1727 %
Root relative squared error             93.0941 %
Total Number of Instances               68

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   Class
  0.682     0          1         0.682     0.811    Y
  1         0.318      0.632     1         0.774    N

=== Confusion Matrix ===

  a  b   <-- classified as
 30 14 |  a = Y
  0 24 |  b = N
```

```
Scheme:       weka.classifiers.bayes.NaiveBayes
Relation:     Image_Features
Instances:    136
Attributes:   23
              File_Name
              Malignant
              Lesion_Nr_Points
              Edge_Nr_Points
              Dir_Changes
              Rays_Sigma
              C_Total
              C_Mean
              C_Sigma
              CD_Total
              CD_Mean
              CD_Sigma
              X_Total
              X_Mean
              X_Sigma
              Y_Total
              Y_Mean
              Y_Sigma
              DC_EP
              TCD_TC
              Cmin
              Cmax
              EP_LP
Test mode:    split 60% train, remainder test
```

```
=== Classifier model (full training set) ===

Naive Bayes Classifier

Class Y: Prior probability = 0.67

File_Name:  Discrete Estimator. Counts =  2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1  (Total = 228)
Lesion_Nr_Points:  Normal Distribution. Mean = 4025.8509 StandardDev =
2562.2427 WeightSum = 92 Precision = 85.59701492537313
Edge_Nr_Points:  Normal Distribution. Mean = 250.3705 StandardDev = 97.2539
WeightSum = 92 Precision = 4.086956521739131
Dir_Changes:  Normal Distribution. Mean = 96.9324 StandardDev = 38.5762
WeightSum = 92 Precision = 2.2222222222222223
Rays_Sigma:  Normal Distribution. Mean = 6.3679 StandardDev = 3.5685
WeightSum = 92 Precision = 0.1331777777777778
C_Total:  Normal Distribution. Mean = 34152562.3894 StandardDev =
117105750.3848 WeightSum = 92 Precision = 6966819.822222223
C_Mean:  Normal Distribution. Mean = 355.337 StandardDev = 103.7689
WeightSum = 92 Precision = 4.25
C_Sigma:  Normal Distribution. Mean = 134.2329 StandardDev = 46.5182
WeightSum = 92 Precision = 1.7259851851851853
CD_Total:  Normal Distribution. Mean = 143567047.6 StandardDev =
365555627.7094 WeightSum = 92 Precision = 1.51123208E7
CD_Mean:  Normal Distribution. Mean = 45.1863 StandardDev = 26.0642
WeightSum = 92 Precision = 1.7142857142857142
CD_Sigma:  Normal Distribution. Mean = 46.4196 StandardDev = 25.1631
WeightSum = 92 Precision = 0.7848923076923077
X_Total:  Normal Distribution. Mean = 37618.1262 StandardDev = 110785.8261
WeightSum = 92 Precision = 5243.738805970149
X_Mean:  Normal Distribution. Mean = 11.952 StandardDev = 5.4169 WeightSum
= 92 Precision = 1.2083333333333333
X_Sigma:  Normal Distribution. Mean = 9.606 StandardDev = 4.0019 WeightSum
```

= 92 Precision = 0.16518796992481205
Y_Total: Normal Distribution. Mean = 33868.653 StandardDev = 111972.5405 WeightSum = 92 Precision = 6424.569230769231
Y_Mean: Normal Distribution. Mean = 10.9447 StandardDev = 4.7645 WeightSum = 92 Precision = 1.0909090909090908
Y_Sigma: Normal Distribution. Mean = 8.7233 StandardDev = 3.5869 WeightSum = 92 Precision = 0.12616666666666665
DC_EP: Normal Distribution. Mean = 0.388 StandardDev = 0.041 WeightSum = 92 Precision = 0.0018078015343511453
TCD_TC: Normal Distribution. Mean = 8.5271 StandardDev = 10.8196 WeightSum = 92 Precision = 0.4691962182518518
Cmin: Normal Distribution. Mean = 221.0083 StandardDev = 103.8611 WeightSum = 92 Precision = 3.847259259259259
Cmax: Normal Distribution. Mean = 489.4982 StandardDev = 122.7524 WeightSum = 92 Precision = 3.945837037037036
EP_LP: Normal Distribution. Mean = 0.0765 StandardDev = 0.0294 WeightSum = 92 Precision = 0.002307840311111111

Class N: Prior probability = 0.33

File_Name: Discrete Estimator. Counts = 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 (Total = 180)
Lesion_Nr_Points: Normal Distribution. Mean = 1496.0024 StandardDev = 1792.7591 WeightSum = 44 Precision = 85.59701492537313
Edge_Nr_Points: Normal Distribution. Mean = 116.8498 StandardDev = 77.5412 WeightSum = 44 Precision = 4.086956521739131
Dir_Changes: Normal Distribution. Mean = 42.2727 StandardDev = 28.0551 WeightSum = 44 Precision = 2.2222222222222223
Rays_Sigma: Normal Distribution. Mean = 2.7392 StandardDev = 1.6928 WeightSum = 44 Precision = 0.1331777777777778
C_Total: Normal Distribution. Mean = 316673.6283 StandardDev = 1451180.8722 WeightSum = 44 Precision = 6966819.822222223

C_Mean: Normal Distribution. Mean = 337.6818 StandardDev = 109.3215 WeightSum = 44 Precision = 4.25
C_Sigma: Normal Distribution. Mean = 75.5511 StandardDev = 41.0738 WeightSum = 44 Precision = 1.7259851851851853
CD_Total: Normal Distribution. Mean = 1030385.5091 StandardDev = 4989071.3536 WeightSum = 44 Precision = 1.51123208E7
CD_Mean: Normal Distribution. Mean = 22.8701 StandardDev = 14.2912 WeightSum = 44 Precision = 1.7142857142857142
CD_Sigma: Normal Distribution. Mean = 22.5657 StandardDev = 15.9931 WeightSum = 44 Precision = 0.7848923076923077
X_Total: Normal Distribution. Mean = 2741.0453 StandardDev = 11804.4298 WeightSum = 44 Precision = 5243.738805970149
X_Mean: Normal Distribution. Mean = 5.108 StandardDev = 3.9312 WeightSum = 44 Precision = 1.2083333333333333
X_Sigma: Normal Distribution. Mean = 3.9683 StandardDev = 3.1003 WeightSum = 44 Precision = 0.16518796992481205
Y_Total: Normal Distribution. Mean = 1022.0906 StandardDev = 4096.1769 WeightSum = 44 Precision = 6424.569230769231
Y_Mean: Normal Distribution. Mean = 4.5124 StandardDev = 3.4229 WeightSum = 44 Precision = 1.0909090909090908
Y_Sigma: Normal Distribution. Mean = 3.7793 StandardDev = 3.0085 WeightSum = 44 Precision = 0.12616666666666665
DC_EP: Normal Distribution. Mean = 0.3669 StandardDev = 0.0528 WeightSum = 44 Precision = 0.0018078015343511453
TCD_TC: Normal Distribution. Mean = 2.3566 StandardDev = 4.5868 WeightSum = 44 Precision = 0.4691962182518518
Cmin: Normal Distribution. Mean = 262.1383 StandardDev = 114.882 WeightSum = 44 Precision = 3.847259259259259
Cmax: Normal Distribution. Mean = 413.1471 StandardDev = 119.3533 WeightSum = 44 Precision = 3.945837037037036
EP_LP: Normal Distribution. Mean = 0.1351 StandardDev = 0.0698 WeightSum = 44 Precision = 0.002307840311111111

Time taken to build model: 0 seconds

```
=== Predictions on test split ===

 inst#,     actual, predicted, error, probability distribution
     1        1:Y        1:Y          *1      0
     2        1:Y        1:Y          *1      0
     3        2:N        2:N           0     *1
     4        1:Y        1:Y          *1      0
     5        2:N        2:N           0     *1
     6        1:Y        1:Y          *1      0
     7        1:Y        2:N      +    0.383 *0.617
     8        1:Y        1:Y          *1      0
     9        2:N        2:N           0     *1
    10        1:Y        2:N      +    0     *1
    11        1:Y        1:Y          *1      0
    12        1:Y        2:N      +    0.248 *0.752
    13        1:Y        1:Y          *1      0
    14        2:N        2:N           0     *1
    15        1:Y        1:Y          *1      0
    16        1:Y        2:N      +    0     *1
    17        2:N        2:N           0     *1
    18        2:N        2:N           0     *1
    19        1:Y        1:Y          *0.998  0.002
    20        2:N        2:N           0     *1
    21        2:N        2:N           0     *1
    22        1:Y        1:Y          *1      0
    23        2:N        2:N           0     *1
    24        1:Y        2:N      +    0     *1
    25        1:Y        2:N      +    0.254 *0.746
    26        1:Y        2:N      +    0     *1
    27        2:N        2:N           0     *1
    28        1:Y        2:N      +    0     *1
    29        2:N        2:N           0     *1
    30        1:Y        1:Y          *1      0
    31        1:Y        1:Y          *1      0
    32        2:N        2:N           0     *1
    33        2:N        2:N           0     *1
```

```
    34        2:N        2:N          0     *1
    35        2:N        2:N          0     *1
    36        1:Y        1:Y         *1      0
    37        1:Y        2:N      +   0.265 *0.735
    38        1:Y        1:Y         *1      0
    39        2:N        2:N          0     *1
    40        2:N        2:N          0     *1
    41        1:Y        1:Y         *1      0
    42        1:Y        1:Y         *1      0
    43        1:Y        1:Y         *1      0
    44        2:N        2:N          0     *1
    45        2:N        2:N          0     *1
    46        1:Y        1:Y         *1      0
    47        1:Y        1:Y         *1      0
    48        1:Y        1:Y         *1      0
    49        1:Y        1:Y         *1      0
    50        1:Y        2:N      +   0.002 *0.998
    51        1:Y        2:N      +   0.117 *0.883
    52        1:Y        1:Y         *1      0
    53        1:Y        1:Y         *1      0
    54        2:N        2:N          0     *1
    55        1:Y        1:Y         *1      0

=== Evaluation on test split ===
=== Summary ===

Correctly Classified Instances          44               80      %
Incorrectly Classified Instances        11               20      %
Kappa statistic                          0.6134
K&B Relative Info Score               3414.6503 %
K&B Information Score                           30.1431 bits        0.5481
bits/instance
Class complexity | order 0                     52.7207 bits        0.9586
bits/instance
Class complexity | scheme                     149.661  bits        2.7211
bits/instance
```

```
Complexity improvement        (Sf)            -96.9403  bits        -1.7626
bits/instance
Mean absolute error                      0.177
Root mean squared error                  0.4004
Relative absolute error                 39.7011 %
Root relative squared error             82.5449 %
Total Number of Instances               55



=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   Class
  0.686     0          1         0.686     0.814    Y
  1         0.314      0.645     1         0.784    N

=== Confusion Matrix ===

  a  b   <-- classified as
 24 11 |  a = Y
  0 20 |  b = N
```

```
Scheme:       weka.classifiers.bayes.NaiveBayes
Relation:     Image_Features
Instances:    136
Attributes:   23
              File_Name
              Malignant
              Lesion_Nr_Points
              Edge_Nr_Points
              Dir_Changes
              Rays_Sigma
              C_Total
              C_Mean
              C_Sigma
              CD_Total
              CD_Mean
              CD_Sigma
              X_Total
              X_Mean
              X_Sigma
              Y_Total
              Y_Mean
              Y_Sigma
              DC_EP
              TCD_TC
              Cmin
              Cmax
              EP_LP
Test mode:    split 70% train, remainder test
```

```
=== Classifier model (full training set) ===

Naive Bayes Classifier

Class Y: Prior probability = 0.67

File_Name:  Discrete Estimator. Counts =  2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1  (Total = 228)
Lesion_Nr_Points:  Normal Distribution. Mean = 4025.8509 StandardDev =
2562.2427 WeightSum = 92 Precision = 85.59701492537313
Edge_Nr_Points:  Normal Distribution. Mean = 250.3705 StandardDev = 97.2539
WeightSum = 92 Precision = 4.086956521739131
Dir_Changes:  Normal Distribution. Mean = 96.9324 StandardDev = 38.5762
WeightSum = 92 Precision = 2.2222222222222223
Rays_Sigma:  Normal Distribution. Mean = 6.3679 StandardDev = 3.5685
WeightSum = 92 Precision = 0.1331777777777778
C_Total:  Normal Distribution. Mean = 34152562.3894 StandardDev =
117105750.3848 WeightSum = 92 Precision = 6966819.822222223
C_Mean:  Normal Distribution. Mean = 355.337 StandardDev = 103.7689
WeightSum = 92 Precision = 4.25
C_Sigma:  Normal Distribution. Mean = 134.2329 StandardDev = 46.5182
WeightSum = 92 Precision = 1.7259851851851853
CD_Total:  Normal Distribution. Mean = 143567047.6 StandardDev =
365555627.7094 WeightSum = 92 Precision = 1.51123208E7
CD_Mean:  Normal Distribution. Mean = 45.1863 StandardDev = 26.0642
WeightSum = 92 Precision = 1.7142857142857142
CD_Sigma:  Normal Distribution. Mean = 46.4196 StandardDev = 25.1631
WeightSum = 92 Precision = 0.7848923076923077
X_Total:  Normal Distribution. Mean = 37618.1262 StandardDev = 110785.8261
WeightSum = 92 Precision = 5243.738805970149
X_Mean:  Normal Distribution. Mean = 11.952 StandardDev = 5.4169 WeightSum
= 92 Precision = 1.2083333333333333
X_Sigma:  Normal Distribution. Mean = 9.606 StandardDev = 4.0019 WeightSum
```

= 92 Precision = 0.16518796992481205
Y_Total: Normal Distribution. Mean = 33868.653 StandardDev = 111972.5405 WeightSum = 92 Precision = 6424.569230769231
Y_Mean: Normal Distribution. Mean = 10.9447 StandardDev = 4.7645 WeightSum = 92 Precision = 1.0909090909090908
Y_Sigma: Normal Distribution. Mean = 8.7233 StandardDev = 3.5869 WeightSum = 92 Precision = 0.12616666666666665
DC_EP: Normal Distribution. Mean = 0.388 StandardDev = 0.041 WeightSum = 92 Precision = 0.0018078015343511453
TCD_TC: Normal Distribution. Mean = 8.5271 StandardDev = 10.8196 WeightSum = 92 Precision = 0.4691962182518518
Cmin: Normal Distribution. Mean = 221.0083 StandardDev = 103.8611 WeightSum = 92 Precision = 3.847259259259259
Cmax: Normal Distribution. Mean = 489.4982 StandardDev = 122.7524 WeightSum = 92 Precision = 3.945837037037036
EP_LP: Normal Distribution. Mean = 0.0765 StandardDev = 0.0294 WeightSum = 92 Precision = 0.002307840311111111

Class N: Prior probability = 0.33

File_Name: Discrete Estimator. Counts = 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 (Total = 180)
Lesion_Nr_Points: Normal Distribution. Mean = 1496.0024 StandardDev = 1792.7591 WeightSum = 44 Precision = 85.59701492537313
Edge_Nr_Points: Normal Distribution. Mean = 116.8498 StandardDev = 77.5412 WeightSum = 44 Precision = 4.086956521739131
Dir_Changes: Normal Distribution. Mean = 42.2727 StandardDev = 28.0551 WeightSum = 44 Precision = 2.2222222222222223
Rays_Sigma: Normal Distribution. Mean = 2.7392 StandardDev = 1.6928 WeightSum = 44 Precision = 0.1331777777777778
C_Total: Normal Distribution. Mean = 316673.6283 StandardDev = 1451180.8722 WeightSum = 44 Precision = 6966819.822222223

C_Mean: Normal Distribution. Mean = 337.6818 StandardDev = 109.3215 WeightSum = 44 Precision = 4.25
C_Sigma: Normal Distribution. Mean = 75.5511 StandardDev = 41.0738 WeightSum = 44 Precision = 1.7259851851851853
CD_Total: Normal Distribution. Mean = 1030385.5091 StandardDev = 4989071.3536 WeightSum = 44 Precision = 1.51123208E7
CD_Mean: Normal Distribution. Mean = 22.8701 StandardDev = 14.2912 WeightSum = 44 Precision = 1.7142857142857142
CD_Sigma: Normal Distribution. Mean = 22.5657 StandardDev = 15.9931 WeightSum = 44 Precision = 0.7848923076923077
X_Total: Normal Distribution. Mean = 2741.0453 StandardDev = 11804.4298 WeightSum = 44 Precision = 5243.738805970149
X_Mean: Normal Distribution. Mean = 5.108 StandardDev = 3.9312 WeightSum = 44 Precision = 1.2083333333333333
X_Sigma: Normal Distribution. Mean = 3.9683 StandardDev = 3.1003 WeightSum = 44 Precision = 0.16518796992481205
Y_Total: Normal Distribution. Mean = 1022.0906 StandardDev = 4096.1769 WeightSum = 44 Precision = 6424.569230769231
Y_Mean: Normal Distribution. Mean = 4.5124 StandardDev = 3.4229 WeightSum = 44 Precision = 1.0909090909090908
Y_Sigma: Normal Distribution. Mean = 3.7793 StandardDev = 3.0085 WeightSum = 44 Precision = 0.12616666666666665
DC_EP: Normal Distribution. Mean = 0.3669 StandardDev = 0.0528 WeightSum = 44 Precision = 0.0018078015343511453
TCD_TC: Normal Distribution. Mean = 2.3566 StandardDev = 4.5868 WeightSum = 44 Precision = 0.4691962182518518
Cmin: Normal Distribution. Mean = 262.1383 StandardDev = 114.882 WeightSum = 44 Precision = 3.847259259259259
Cmax: Normal Distribution. Mean = 413.1471 StandardDev = 119.3533 WeightSum = 44 Precision = 3.945837037037036
EP_LP: Normal Distribution. Mean = 0.1351 StandardDev = 0.0698 WeightSum = 44 Precision = 0.002307840311111111

Time taken to build model: 0 seconds

```
=== Predictions on test split ===

 inst#,     actual, predicted, error, probability distribution
      1        1:Y        1:Y          *1      0
      2        1:Y        2:N      +   0      *1
      3        2:N        2:N           0      *1
      4        2:N        2:N           0      *1
      5        1:Y        1:Y          *0.999  0.001
      6        2:N        2:N           0      *1
      7        2:N        2:N           0      *1
      8        1:Y        1:Y          *1      0
      9        2:N        2:N           0      *1
     10        1:Y        2:N      +   0      *1
     11        1:Y        1:Y          *0.918  0.082
     12        1:Y        2:N      +   0      *1
     13        2:N        2:N           0      *1
     14        1:Y        2:N      +   0      *1
     15        2:N        2:N           0      *1
     16        1:Y        1:Y          *1      0
     17        1:Y        1:Y          *1      0
     18        2:N        2:N           0      *1
     19        2:N        2:N           0      *1
     20        2:N        2:N           0      *1
     21        2:N        2:N           0      *1
     22        1:Y        1:Y          *1      0
     23        1:Y        2:N      +   0.215  *0.785
     24        1:Y        1:Y          *1      0
     25        2:N        2:N           0      *1
     26        2:N        2:N           0      *1
     27        1:Y        1:Y          *1      0
     28        1:Y        1:Y          *1      0
     29        1:Y        1:Y          *1      0
     30        2:N        2:N           0      *1
     31        2:N        2:N           0      *1
     32        1:Y        1:Y          *1      0
     33        1:Y        1:Y          *1      0
```

```
    34        1:Y        1:Y          *1      0
    35        1:Y        1:Y          *1      0
    36        1:Y        2:N      +   0.013 *0.987
    37        1:Y        2:N      +   0.457 *0.543
    38        1:Y        1:Y         *0.991  0.009
    39        1:Y        1:Y          *1      0
    40        2:N        2:N          0.001 *0.999
    41        1:Y        1:Y          *1      0

=== Evaluation on test split ===
=== Summary ===

Correctly Classified Instances          34               82.9268 %
Incorrectly Classified Instances         7               17.0732 %
Kappa statistic                          0.6674
K&B Relative Info Score               2955.6662 %
K&B Information Score                   26.0107 bits        0.6344
bits/instance
Class complexity | order 0              40.6822 bits        0.9922
bits/instance
Class complexity | scheme              112.4906 bits        2.7437
bits/instance
Complexity improvement        (Sf)     -71.8084 bits       -1.7514
bits/instance
Mean absolute error                      0.1563
Root mean squared error                  0.3791
Relative absolute error                 34.2866 %
Root relative squared error             76.3847 %
Total Number of Instances               41

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   Class
  0.72      0          1         0.72      0.837    Y
  1         0.28       0.696     1         0.821    N
```

```
=== Confusion Matrix ===

  a  b   <-- classified as
 18  7 |  a = Y
  0 16 |  b = N
```

```
Scheme:       weka.classifiers.bayes.NaiveBayes
Relation:     Image_Features
Instances:    136
Attributes:   23
              File_Name
              Malignant
              Lesion_Nr_Points
              Edge_Nr_Points
              Dir_Changes
              Rays_Sigma
              C_Total
              C_Mean
              C_Sigma
              CD_Total
              CD_Mean
              CD_Sigma
              X_Total
              X_Mean
              X_Sigma
              Y_Total
              Y_Mean
              Y_Sigma
              DC_EP
              TCD_TC
              Cmin
              Cmax
              EP_LP
Test mode:    split 80% train, remainder test
```

```
=== Classifier model (full training set) ===

Naive Bayes Classifier

Class Y: Prior probability = 0.67

File_Name:  Discrete Estimator. Counts =  2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1  (Total = 228)
Lesion_Nr_Points:  Normal Distribution. Mean = 4025.8509 StandardDev =
2562.2427 WeightSum = 92 Precision = 85.59701492537313
Edge_Nr_Points:  Normal Distribution. Mean = 250.3705 StandardDev = 97.2539
WeightSum = 92 Precision = 4.086956521739131
Dir_Changes:  Normal Distribution. Mean = 96.9324 StandardDev = 38.5762
WeightSum = 92 Precision = 2.2222222222222223
Rays_Sigma:  Normal Distribution. Mean = 6.3679 StandardDev = 3.5685
WeightSum = 92 Precision = 0.1331777777777778
C_Total:  Normal Distribution. Mean = 34152562.3894 StandardDev =
117105750.3848 WeightSum = 92 Precision = 6966819.822222223
C_Mean:  Normal Distribution. Mean = 355.337 StandardDev = 103.7689
WeightSum = 92 Precision = 4.25
C_Sigma:  Normal Distribution. Mean = 134.2329 StandardDev = 46.5182
WeightSum = 92 Precision = 1.7259851851851853
CD_Total:  Normal Distribution. Mean = 143567047.6 StandardDev =
365555627.7094 WeightSum = 92 Precision = 1.51123208E7
CD_Mean:  Normal Distribution. Mean = 45.1863 StandardDev = 26.0642
WeightSum = 92 Precision = 1.7142857142857142
CD_Sigma:  Normal Distribution. Mean = 46.4196 StandardDev = 25.1631
WeightSum = 92 Precision = 0.7848923076923077
X_Total:  Normal Distribution. Mean = 37618.1262 StandardDev = 110785.8261
WeightSum = 92 Precision = 5243.738805970149
X_Mean:  Normal Distribution. Mean = 11.952 StandardDev = 5.4169 WeightSum
= 92 Precision = 1.2083333333333333
X_Sigma:  Normal Distribution. Mean = 9.606 StandardDev = 4.0019 WeightSum
```

= 92 Precision = 0.16518796992481205
Y_Total: Normal Distribution. Mean = 33868.653 StandardDev = 111972.5405 WeightSum = 92 Precision = 6424.569230769231
Y_Mean: Normal Distribution. Mean = 10.9447 StandardDev = 4.7645 WeightSum = 92 Precision = 1.0909090909090908
Y_Sigma: Normal Distribution. Mean = 8.7233 StandardDev = 3.5869 WeightSum = 92 Precision = 0.12616666666666665
DC_EP: Normal Distribution. Mean = 0.388 StandardDev = 0.041 WeightSum = 92 Precision = 0.0018078015343511453
TCD_TC: Normal Distribution. Mean = 8.5271 StandardDev = 10.8196 WeightSum = 92 Precision = 0.4691962182518518
Cmin: Normal Distribution. Mean = 221.0083 StandardDev = 103.8611 WeightSum = 92 Precision = 3.847259259259259
Cmax: Normal Distribution. Mean = 489.4982 StandardDev = 122.7524 WeightSum = 92 Precision = 3.945837037037036
EP_LP: Normal Distribution. Mean = 0.0765 StandardDev = 0.0294 WeightSum = 92 Precision = 0.002307840311111111

Class N: Prior probability = 0.33

File_Name: Discrete Estimator. Counts = 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 (Total = 180)
Lesion_Nr_Points: Normal Distribution. Mean = 1496.0024 StandardDev = 1792.7591 WeightSum = 44 Precision = 85.59701492537313
Edge_Nr_Points: Normal Distribution. Mean = 116.8498 StandardDev = 77.5412 WeightSum = 44 Precision = 4.086956521739131
Dir_Changes: Normal Distribution. Mean = 42.2727 StandardDev = 28.0551 WeightSum = 44 Precision = 2.2222222222222223
Rays_Sigma: Normal Distribution. Mean = 2.7392 StandardDev = 1.6928 WeightSum = 44 Precision = 0.1331777777777778
C_Total: Normal Distribution. Mean = 316673.6283 StandardDev = 1451180.8722 WeightSum = 44 Precision = 6966819.822222223

C_Mean: Normal Distribution. Mean = 337.6818 StandardDev = 109.3215 WeightSum = 44 Precision = 4.25
C_Sigma: Normal Distribution. Mean = 75.5511 StandardDev = 41.0738 WeightSum = 44 Precision = 1.7259851851851853
CD_Total: Normal Distribution. Mean = 1030385.5091 StandardDev = 4989071.3536 WeightSum = 44 Precision = 1.51123208E7
CD_Mean: Normal Distribution. Mean = 22.8701 StandardDev = 14.2912 WeightSum = 44 Precision = 1.7142857142857142
CD_Sigma: Normal Distribution. Mean = 22.5657 StandardDev = 15.9931 WeightSum = 44 Precision = 0.7848923076923077
X_Total: Normal Distribution. Mean = 2741.0453 StandardDev = 11804.4298 WeightSum = 44 Precision = 5243.738805970149
X_Mean: Normal Distribution. Mean = 5.108 StandardDev = 3.9312 WeightSum = 44 Precision = 1.2083333333333333
X_Sigma: Normal Distribution. Mean = 3.9683 StandardDev = 3.1003 WeightSum = 44 Precision = 0.16518796992481205
Y_Total: Normal Distribution. Mean = 1022.0906 StandardDev = 4096.1769 WeightSum = 44 Precision = 6424.569230769231
Y_Mean: Normal Distribution. Mean = 4.5124 StandardDev = 3.4229 WeightSum = 44 Precision = 1.0909090909090908
Y_Sigma: Normal Distribution. Mean = 3.7793 StandardDev = 3.0085 WeightSum = 44 Precision = 0.12616666666666665
DC_EP: Normal Distribution. Mean = 0.3669 StandardDev = 0.0528 WeightSum = 44 Precision = 0.0018078015343511453
TCD_TC: Normal Distribution. Mean = 2.3566 StandardDev = 4.5868 WeightSum = 44 Precision = 0.4691962182518518
Cmin: Normal Distribution. Mean = 262.1383 StandardDev = 114.882 WeightSum = 44 Precision = 3.847259259259259
Cmax: Normal Distribution. Mean = 413.1471 StandardDev = 119.3533 WeightSum = 44 Precision = 3.945837037037036
EP_LP: Normal Distribution. Mean = 0.1351 StandardDev = 0.0698 WeightSum = 44 Precision = 0.002307840311111111

Time taken to build model: 0 seconds

```
=== Predictions on test split ===

 inst#,     actual, predicted, error, probability distribution
     1        1:Y        2:N      +   0     *1
     2        2:N        2:N          0     *1
     3        1:Y        1:Y         *1      0
     4        1:Y        1:Y         *1      0
     5        2:N        2:N          0     *1
     6        2:N        2:N          0     *1
     7        2:N        2:N          0     *1
     8        2:N        2:N          0     *1
     9        1:Y        1:Y         *1      0
    10        1:Y        2:N      +   0.201 *0.799
    11        1:Y        1:Y         *1      0
    12        2:N        2:N          0     *1
    13        2:N        2:N          0     *1
    14        1:Y        1:Y         *1      0
    15        1:Y        1:Y         *1      0
    16        1:Y        1:Y         *1      0
    17        2:N        2:N          0     *1
    18        2:N        2:N          0     *1
    19        1:Y        1:Y         *1      0
    20        1:Y        1:Y         *1      0
    21        1:Y        1:Y         *1      0
    22        1:Y        1:Y         *1      0
    23        1:Y        2:N      +   0.086 *0.914
    24        1:Y        2:N      +   0.436 *0.564
    25        1:Y        1:Y         *0.998  0.002
    26        1:Y        1:Y         *1      0
    27        2:N        2:N          0     *1
    28        1:Y        1:Y         *1      0
```

```
=== Evaluation on test split ===
=== Summary ===

Correctly Classified Instances          24               85.7143 %
Incorrectly Classified Instances         4               14.2857 %
Kappa statistic                          0.7143
K&B Relative Info Score               2096.8376 %
K&B Information Score                   18.9217 bits        0.6758
bits/instance
Class complexity | order 0              26.4665 bits        0.9452
bits/instance
Class complexity | scheme               36.6841 bits        1.3101
bits/instance
Complexity improvement        (Sf)      -10.2175 bits       -0.3649
bits/instance
Mean absolute error                      0.1172
Root mean squared error                  0.3158
Relative absolute error                 26.1508 %
Root relative squared error             65.699  %
Total Number of Instances               28

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   Class
  0.778     0          1         0.778     0.875    Y
  1         0.222      0.714     1         0.833    N

=== Confusion Matrix ===

  a  b   <-- classified as
 14  4 |  a = Y
  0 10 |  b = N
```

```
Scheme:                       weka.classifiers.bayes.BayesNet -D -Q
weka.classifiers.bayes.net.search.local.TAN -- -S BAYES -E
weka.classifiers.bayes.net.estimate.SimpleEstimator -- -A 0.5
Relation:     Image_Features
Instances:    136
Attributes:   23
              File_Name
              Malignant
              Lesion_Nr_Points
              Edge_Nr_Points
              Dir_Changes
              Rays_Sigma
              C_Total
              C_Mean
              C_Sigma
              CD_Total
              CD_Mean
              CD_Sigma
              X_Total
              X_Mean
              X_Sigma
              Y_Total
              Y_Mean
              Y_Sigma
              DC_EP
              TCD_TC
              Cmin
              Cmax
              EP_LP
Test mode:    split 50% train, remainder test
```

```
=== Classifier model (full training set) ===

Bayes Network Classifier
not using ADTree
#attributes=23 #classindex=1
Network structure (nodes followed by parents)
File_Name(136): Malignant Malignant CD_Mean
Malignant(2):
Lesion_Nr_Points(2): Malignant Malignant Dir_Changes
Edge_Nr_Points(2): Malignant Malignant Dir_Changes
Dir_Changes(2): Malignant Malignant Y_Total
Rays_Sigma(2): Malignant Malignant File_Name
C_Total(3): Malignant Malignant File_Name
C_Mean(1): Malignant Malignant File_Name
C_Sigma(2): Malignant Malignant CD_Total
CD_Total(3): Malignant Malignant File_Name
CD_Mean(2): Malignant Malignant CD_Sigma
CD_Sigma(2): Malignant Malignant TCD_TC
X_Total(2): Malignant Malignant C_Total
X_Mean(2): Malignant Malignant X_Total
X_Sigma(2): Malignant Malignant X_Mean
Y_Total(2): Malignant Malignant C_Total
Y_Mean(3): Malignant Malignant File_Name
Y_Sigma(2): Malignant Malignant Y_Mean
DC_EP(1): Malignant Malignant File_Name
TCD_TC(2): Malignant Malignant
Cmin(1): Malignant Malignant File_Name
Cmax(1): Malignant Malignant File_Name
EP_LP(2): Malignant Malignant Lesion_Nr_Points
LogScore Bayes: -1796.618501744451
LogScore BDeu: -37288.979515296414
LogScore MDL: -19579.89330778845
LogScore ENTROPY: -7285.974456234084
LogScore AIC: -12290.974456234087

Time taken to build model: 0.12 seconds
```

```
=== Predictions on test split ===

 inst#,     actual, predicted, error, probability distribution
     1        1:Y        1:Y          *0.985  0.015
     2        2:N        2:N           0.095 *0.905
     3        1:Y        2:N      +    0.345 *0.655
     4        1:Y        2:N      +    0.453 *0.547
     5        2:N        1:Y      +   *0.904  0.096
     6        2:N        2:N           0.001 *0.999
     7        2:N        1:Y      +   *0.813  0.187
     8        1:Y        1:Y          *0.922  0.078
     9        1:Y        2:N      +    0.001 *0.999
    10        1:Y        1:Y          *0.985  0.015
    11        1:Y        1:Y          *0.558  0.442
    12        1:Y        1:Y          *0.797  0.203
    13        1:Y        1:Y          *0.995  0.005
    14        1:Y        1:Y          *0.997  0.003
    15        1:Y        1:Y          *0.99   0.01
    16        2:N        1:Y      +   *0.719  0.281
    17        1:Y        1:Y          *0.995  0.005
    18        2:N        2:N           0.013 *0.987
    19        1:Y        1:Y          *0.99   0.01
    20        1:Y        1:Y          *0.813  0.187
    21        1:Y        1:Y          *0.993  0.007
    22        2:N        2:N           0     *1
    23        1:Y        2:N      +    0.236 *0.764
    24        1:Y        1:Y          *0.99   0.01
    25        1:Y        2:N      +    0.478 *0.522
    26        1:Y        1:Y          *0.984  0.016
    27        2:N        2:N           0.001 *0.999
    28        1:Y        1:Y          *0.973  0.027
    29        1:Y        2:N      +    0.274 *0.726
    30        2:N        2:N           0     *1
    31        2:N        1:Y      +   *0.955  0.045
    32        1:Y        1:Y          *0.99   0.01
    33        2:N        2:N           0.274 *0.726
```

```
34       2:N       2:N          0.001 *0.999
35       1:Y       1:Y        *0.99   0.01
36       2:N       2:N          0.001 *0.999
37       1:Y       2:N      +   0.007 *0.993
38       1:Y       1:Y        *0.99   0.01
39       1:Y       1:Y        *0.813  0.187
40       2:N       1:Y      +  *0.719  0.281
41       1:Y       2:N      +   0.044 *0.956
42       2:N       2:N          0     *1
43       1:Y       1:Y        *0.99   0.01
44       1:Y       1:Y        *0.99   0.01
45       2:N       1:Y      +  *0.689  0.311
46       2:N       2:N          0     *1
47       2:N       2:N          0.035 *0.965
48       2:N       1:Y      +  *0.717  0.283
49       1:Y       1:Y        *0.99   0.01
50       1:Y       1:Y        *0.99   0.01
51       1:Y       1:Y        *0.99   0.01
52       2:N       2:N          0.001 *0.999
53       2:N       2:N          0.06  *0.94
54       1:Y       1:Y        *0.99   0.01
55       1:Y       1:Y        *0.99   0.01
56       1:Y       1:Y        *0.985  0.015
57       2:N       2:N          0.345 *0.655
58       2:N       2:N          0.04  *0.96
59       1:Y       1:Y        *0.985  0.015
60       1:Y       1:Y        *0.99   0.01
61       1:Y       1:Y        *0.922  0.078
62       1:Y       1:Y        *0.99   0.01
63       1:Y       1:Y        *0.813  0.187
64       1:Y       2:N      +   0.274 *0.726
65       1:Y       1:Y        *0.996  0.004
66       1:Y       1:Y        *0.993  0.007
67       2:N       2:N          0.257 *0.743
68       1:Y       1:Y        *0.99   0.01
```

```
=== Evaluation on test split ===
=== Summary ===

Correctly Classified Instances          52               76.4706 %
Incorrectly Classified Instances        16               23.5294 %
Kappa statistic                          0.4944
K&B Relative Info Score               3380.8756 %
K&B Information Score                   29.7953 bits        0.4382
bits/instance
Class complexity | order 0              64.3284 bits         0.946
bits/instance
Class complexity | scheme               54.02    bits        0.7944
bits/instance
Complexity improvement       (Sf)       10.3084 bits         0.1516
bits/instance
Mean absolute error                      0.2233
Root mean squared error                  0.3945
Relative absolute error                 50.6068 %
Root relative squared error             82.0471 %
Total Number of Instances               68


=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   Class
  0.795     0.292      0.833     0.795     0.814    Y
  0.708     0.205      0.654     0.708     0.68     N

=== Confusion Matrix ===

  a  b   <-- classified as
 35  9 |  a = Y
  7 17 |  b = N
```

```
Scheme:                       weka.classifiers.bayes.BayesNet -D -Q
weka.classifiers.bayes.net.search.local.TAN -- -S BAYES -E
weka.classifiers.bayes.net.estimate.SimpleEstimator -- -A 0.5
Relation:     Image_Features
Instances:    136
Attributes:   23
              File_Name
              Malignant
              Lesion_Nr_Points
              Edge_Nr_Points
              Dir_Changes
              Rays_Sigma
              C_Total
              C_Mean
              C_Sigma
              CD_Total
              CD_Mean
              CD_Sigma
              X_Total
              X_Mean
              X_Sigma
              Y_Total
              Y_Mean
              Y_Sigma
              DC_EP
              TCD_TC
              Cmin
              Cmax
              EP_LP
Test mode:    split 60% train, remainder test

=== Classifier model (full training set) ===

Bayes Network Classifier
not using ADTree
#attributes=23 #classindex=1
```

```
Network structure (nodes followed by parents)
File_Name(136): Malignant Malignant CD_Mean
Malignant(2):
Lesion_Nr_Points(2): Malignant Malignant Dir_Changes
Edge_Nr_Points(2): Malignant Malignant Dir_Changes
Dir_Changes(2): Malignant Malignant Y_Total
Rays_Sigma(2): Malignant Malignant File_Name
C_Total(3): Malignant Malignant File_Name
C_Mean(1): Malignant Malignant File_Name
C_Sigma(2): Malignant Malignant CD_Total
CD_Total(3): Malignant Malignant File_Name
CD_Mean(2): Malignant Malignant CD_Sigma
CD_Sigma(2): Malignant Malignant TCD_TC
X_Total(2): Malignant Malignant C_Total
X_Mean(2): Malignant Malignant X_Total
X_Sigma(2): Malignant Malignant X_Mean
Y_Total(2): Malignant Malignant C_Total
Y_Mean(3): Malignant Malignant File_Name
Y_Sigma(2): Malignant Malignant Y_Mean
DC_EP(1): Malignant Malignant File_Name
TCD_TC(2): Malignant Malignant
Cmin(1): Malignant Malignant File_Name
Cmax(1): Malignant Malignant File_Name
EP_LP(2): Malignant Malignant Lesion_Nr_Points
LogScore Bayes: -1796.618501744451
LogScore BDeu: -37288.979515296414
LogScore MDL: -19579.89330778845
LogScore ENTROPY: -7285.974456234084
LogScore AIC: -12290.974456234087


Time taken to build model: 0.12 seconds
```

```
=== Predictions on test split ===

 inst#,     actual, predicted, error, probability distribution
     1        1:Y        1:Y          *0.957  0.043
     2        1:Y        1:Y          *0.957  0.043
     3        2:N        2:N           0.143 *0.857
     4        1:Y        1:Y          *0.972  0.028
     5        2:N        2:N           0.002 *0.998
     6        1:Y        1:Y          *0.957  0.043
     7        1:Y        1:Y          *0.803  0.197
     8        1:Y        1:Y          *0.967  0.033
     9        2:N        2:N           0.001 *0.999
    10        1:Y        1:Y          *0.791  0.209
    11        1:Y        1:Y          *0.957  0.043
    12        1:Y        1:Y          *0.909  0.091
    13        1:Y        1:Y          *0.89   0.11
    14        2:N        2:N           0.005 *0.995
    15        1:Y        1:Y          *0.86   0.14
    16        1:Y        1:Y          *0.766  0.234
    17        2:N        2:N           0.001 *0.999
    18        2:N        1:Y      +   *0.993  0.007
    19        1:Y        1:Y          *0.957  0.043
    20        2:N        1:Y      +   *0.766  0.234
    21        2:N        2:N           0.005 *0.995
    22        1:Y        1:Y          *0.957  0.043
    23        2:N        2:N           0.14  *0.86
    24        1:Y        2:N      +    0.14  *0.86
    25        1:Y        1:Y          *0.957  0.043
    26        1:Y        1:Y          *0.803  0.197
    27        2:N        2:N           0.143 *0.857
    28        1:Y        2:N      +    0.474 *0.526
    29        2:N        2:N           0.001 *0.999
    30        1:Y        1:Y          *0.957  0.043
    31        1:Y        1:Y          *0.957  0.043
    32        2:N        2:N           0.425 *0.575
    33        2:N        2:N           0.001 *0.999
```

```
    34        2:N        2:N          0.108 *0.892
    35        2:N        2:N          0.148 *0.852
    36        1:Y        1:Y         *0.957  0.043
    37        1:Y        1:Y         *0.957  0.043
    38        1:Y        1:Y         *0.957  0.043
    39        2:N        2:N          0.034 *0.966
    40        2:N        1:Y      +  *0.627  0.373
    41        1:Y        1:Y         *0.957  0.043
    42        1:Y        1:Y         *0.957  0.043
    43        1:Y        1:Y         *0.73   0.27
    44        2:N        1:Y      +  *0.529  0.471
    45        2:N        1:Y      +  *0.514  0.486
    46        1:Y        1:Y         *0.635  0.365
    47        1:Y        1:Y         *0.957  0.043
    48        1:Y        1:Y         *0.86   0.14
    49        1:Y        1:Y         *0.957  0.043
    50        1:Y        1:Y         *0.766  0.234
    51        1:Y        1:Y         *0.766  0.234
    52        1:Y        1:Y         *0.967  0.033
    53        1:Y        1:Y         *0.967  0.033
    54        2:N        1:Y      +  *0.766  0.234
    55        1:Y        1:Y         *0.957  0.043

=== Evaluation on test split ===
=== Summary ===

Correctly Classified Instances          47               85.4545 %
Incorrectly Classified Instances         8               14.5455 %
Kappa statistic                          0.6716
K&B Relative Info Score               3686.2948 %
K&B Information Score                            32.5411 bits        0.5917
bits/instance
Class complexity | order 0                       52.7207 bits        0.9586
bits/instance
Class complexity | scheme                        25.9923 bits        0.4726
bits/instance
```

```
Complexity improvement        (Sf)              26.7284 bits         0.486
bits/instance
Mean absolute error                      0.1822
Root mean squared error                  0.3008
Relative absolute error                 40.861  %
Root relative squared error             62.0051 %
Total Number of Instances               55

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   Class
  0.943     0.3        0.846     0.943     0.892    Y
  0.7       0.057      0.875     0.7       0.778    N

=== Confusion Matrix ===

  a  b   <-- classified as
 33  2 |  a = Y
  6 14 |  b = N
```

```
Scheme:                         weka.classifiers.bayes.BayesNet    -D    -Q
weka.classifiers.bayes.net.search.local.TAN     --      -S      BAYES      -E
weka.classifiers.bayes.net.estimate.SimpleEstimator -- -A 0.5
Relation:     Image_Features
Instances:    136
Attributes:   23
              File_Name
              Malignant
              Lesion_Nr_Points
              Edge_Nr_Points
              Dir_Changes
              Rays_Sigma
              C_Total
              C_Mean
              C_Sigma
              CD_Total
              CD_Mean
              CD_Sigma
              X_Total
              X_Mean
              X_Sigma
              Y_Total
              Y_Mean
              Y_Sigma
              DC_EP
              TCD_TC
              Cmin
              Cmax
              EP_LP
Test mode:    split 70% train, remainder test

=== Classifier model (full training set) ===

Bayes Network Classifier
not using ADTree
#attributes=23 #classindex=1
```

```
Network structure (nodes followed by parents)
File_Name(136): Malignant Malignant CD_Mean
Malignant(2):
Lesion_Nr_Points(2): Malignant Malignant Dir_Changes
Edge_Nr_Points(2): Malignant Malignant Dir_Changes
Dir_Changes(2): Malignant Malignant Y_Total
Rays_Sigma(2): Malignant Malignant File_Name
C_Total(3): Malignant Malignant File_Name
C_Mean(1): Malignant Malignant File_Name
C_Sigma(2): Malignant Malignant CD_Total
CD_Total(3): Malignant Malignant File_Name
CD_Mean(2): Malignant Malignant CD_Sigma
CD_Sigma(2): Malignant Malignant TCD_TC
X_Total(2): Malignant Malignant C_Total
X_Mean(2): Malignant Malignant X_Total
X_Sigma(2): Malignant Malignant X_Mean
Y_Total(2): Malignant Malignant C_Total
Y_Mean(3): Malignant Malignant File_Name
Y_Sigma(2): Malignant Malignant Y_Mean
DC_EP(1): Malignant Malignant File_Name
TCD_TC(2): Malignant Malignant
Cmin(1): Malignant Malignant File_Name
Cmax(1): Malignant Malignant File_Name
EP_LP(2): Malignant Malignant Lesion_Nr_Points
LogScore Bayes: -1796.618501744451
LogScore BDeu: -37288.979515296414
LogScore MDL: -19579.89330778845
LogScore ENTROPY: -7285.974456234084
LogScore AIC: -12290.974456234087


Time taken to build model: 0.08 seconds
```

```
=== Predictions on test split ===

 inst#,     actual, predicted, error, probability distribution
     1        1:Y        1:Y          *0.91   0.09
     2        1:Y        1:Y          *0.687  0.313
     3        2:N        2:N           0.005 *0.995
     4        2:N        1:Y      +   *0.997  0.003
     5        1:Y        1:Y          *0.974  0.026
     6        2:N        1:Y      +   *0.687  0.313
     7        2:N        2:N           0.029 *0.971
     8        1:Y        1:Y          *0.974  0.026
     9        2:N        2:N           0.216 *0.784
    10        1:Y        2:N      +    0.216 *0.784
    11        1:Y        1:Y          *0.974  0.026
    12        1:Y        1:Y          *0.761  0.239
    13        2:N        2:N           0.096 *0.904
    14        1:Y        2:N      +    0.372 *0.628
    15        2:N        2:N           0.005 *0.995
    16        1:Y        1:Y          *0.974  0.026
    17        1:Y        1:Y          *0.974  0.026
    18        2:N        1:Y      +   *0.597  0.403
    19        2:N        2:N           0.005 *0.995
    20        2:N        2:N           0.046 *0.954
    21        2:N        2:N           0.093 *0.907
    22        1:Y        1:Y          *0.974  0.026
    23        1:Y        1:Y          *0.974  0.026
    24        1:Y        1:Y          *0.974  0.026
    25        2:N        2:N           0.01  *0.99
    26        2:N        2:N           0.349 *0.651
    27        1:Y        1:Y          *0.974  0.026
    28        1:Y        1:Y          *0.974  0.026
    29        1:Y        2:N      +    0.299 *0.701
    30        2:N        2:N           0.334 *0.666
    31        2:N        2:N           0.376 *0.624
    32        1:Y        1:Y          *0.687  0.313
    33        1:Y        1:Y          *0.974  0.026
```

```
    34        1:Y        1:Y          *0.91   0.09
    35        1:Y        1:Y          *0.974  0.026
    36        1:Y        1:Y          *0.687  0.313
    37        1:Y        1:Y          *0.687  0.313
    38        1:Y        1:Y          *0.974  0.026
    39        1:Y        1:Y          *0.974  0.026
    40        2:N        1:Y      +   *0.687  0.313
    41        1:Y        1:Y          *0.974  0.026

=== Evaluation on test split ===
=== Summary ===

Correctly Classified Instances          34               82.9268 %
Incorrectly Classified Instances         7               17.0732 %
Kappa statistic                          0.6372
K&B Relative Info Score               2537.505  %
K&B  Information  Score                          22.3307  bits        0.5447
bits/instance
Class  complexity  |  order  0                   40.6822  bits        0.9922
bits/instance
Class  complexity  |  scheme                     24.7223  bits         0.603
bits/instance
Complexity  improvement        (Sf)             15.9599  bits        0.3893
bits/instance
Mean absolute error                      0.2122
Root mean squared error                  0.339
Relative absolute error                 46.5509 %
Root relative squared error             68.309  %
Total Number of Instances               41

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   Class
  0.88      0.25       0.846     0.88      0.863    Y
  0.75      0.12       0.8       0.75      0.774    N
```

```
=== Confusion Matrix ===

  a  b   <-- classified as
 22  3 |  a = Y
  4 12 |  b = N
```

```
Scheme:                       weka.classifiers.bayes.BayesNet -D -Q
weka.classifiers.bayes.net.search.local.TAN -- -S BAYES -E
weka.classifiers.bayes.net.estimate.SimpleEstimator -- -A 0.5
Relation:     Image_Features
Instances:    136
Attributes:   23
              File_Name
              Malignant
              Lesion_Nr_Points
              Edge_Nr_Points
              Dir_Changes
              Rays_Sigma
              C_Total
              C_Mean
              C_Sigma
              CD_Total
              CD_Mean
              CD_Sigma
              X_Total
              X_Mean
              X_Sigma
              Y_Total
              Y_Mean
              Y_Sigma
              DC_EP
              TCD_TC
              Cmin
              Cmax
              EP_LP
Test mode:    split 80% train, remainder test
```

```
=== Classifier model (full training set) ===

Bayes Network Classifier
not using ADTree
#attributes=23 #classindex=1
Network structure (nodes followed by parents)
File_Name(136): Malignant Malignant CD_Mean
Malignant(2):
Lesion_Nr_Points(2): Malignant Malignant Dir_Changes
Edge_Nr_Points(2): Malignant Malignant Dir_Changes
Dir_Changes(2): Malignant Malignant Y_Total
Rays_Sigma(2): Malignant Malignant File_Name
C_Total(3): Malignant Malignant File_Name
C_Mean(1): Malignant Malignant File_Name
C_Sigma(2): Malignant Malignant CD_Total
CD_Total(3): Malignant Malignant File_Name
CD_Mean(2): Malignant Malignant CD_Sigma
CD_Sigma(2): Malignant Malignant TCD_TC
X_Total(2): Malignant Malignant C_Total
X_Mean(2): Malignant Malignant X_Total
X_Sigma(2): Malignant Malignant X_Mean
Y_Total(2): Malignant Malignant C_Total
Y_Mean(3): Malignant Malignant File_Name
Y_Sigma(2): Malignant Malignant Y_Mean
DC_EP(1): Malignant Malignant File_Name
TCD_TC(2): Malignant Malignant
Cmin(1): Malignant Malignant File_Name
Cmax(1): Malignant Malignant File_Name
EP_LP(2): Malignant Malignant Lesion_Nr_Points
LogScore Bayes: -1796.618501744451
LogScore BDeu: -37288.979515296414
LogScore MDL: -19579.89330778845
LogScore ENTROPY: -7285.974456234084
LogScore AIC: -12290.974456234087

Time taken to build model: 0.08 seconds
```

```
=== Predictions on test split ===

 inst#,     actual, predicted, error, probability distribution
      1        1:Y        2:N      +   0.229 *0.771
      2        2:N        2:N          0     *1
      3        1:Y        1:Y         *0.993  0.007
      4        1:Y        1:Y         *0.993  0.007
      5        2:N        2:N          0.461 *0.539
      6        2:N        2:N          0     *1
      7        2:N        2:N          0.225 *0.775
      8        2:N        2:N          0.032 *0.968
      9        1:Y        1:Y         *0.993  0.007
     10        1:Y        1:Y         *0.993  0.007
     11        1:Y        1:Y         *0.993  0.007
     12        2:N        2:N          0.003 *0.997
     13        2:N        2:N          0.493 *0.507
     14        1:Y        1:Y         *0.993  0.007
     15        1:Y        1:Y         *0.993  0.007
     16        1:Y        1:Y         *0.906  0.094
     17        2:N        2:N          0.489 *0.511
     18        2:N        2:N          0.028 *0.972
     19        1:Y        1:Y         *0.877  0.123
     20        1:Y        1:Y         *0.993  0.007
     21        1:Y        1:Y         *0.977  0.023
     22        1:Y        1:Y         *0.993  0.007
     23        1:Y        1:Y         *0.506  0.494
     24        1:Y        1:Y         *0.506  0.494
     25        1:Y        1:Y         *0.993  0.007
     26        1:Y        1:Y         *0.993  0.007
     27        2:N        1:Y      +  *0.506  0.494
     28        1:Y        1:Y         *0.993  0.007
```

```
=== Evaluation on test split ===
=== Summary ===

Correctly Classified Instances          26               92.8571 %
Incorrectly Classified Instances         2                7.1429 %
Kappa statistic                          0.8444
K&B Relative Info Score               1935.051  %
K&B Information Score                   17.4618  bits        0.6236
bits/instance
Class complexity | order 0              26.4665  bits        0.9452
bits/instance
Class complexity | scheme                8.8943  bits        0.3177
bits/instance
Complexity improvement       (Sf)       17.5722  bits        0.6276
bits/instance
Mean absolute error                      0.1542
Root mean squared error                  0.2746
Relative absolute error                 34.41   %
Root relative squared error             57.1242 %
Total Number of Instances               28

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   Class
  0.944     0.1        0.944     0.944     0.944    Y
  0.9       0.056      0.9       0.9       0.9      N

=== Confusion Matrix ===

  a  b   <-- classified as
 17  1 |  a = Y
  1  9 |  b = N
```

```
Scheme:       weka.classifiers.lazy.IBk -K 9 -W 0
Relation:     Image_Features
Instances:    136
Attributes:   23
              File_Name
              Malignant
              Lesion_Nr_Points
              Edge_Nr_Points
              Dir_Changes
              Rays_Sigma
              C_Total
              C_Mean
              C_Sigma
              CD_Total
              CD_Mean
              CD_Sigma
              X_Total
              X_Mean
              X_Sigma
              Y_Total
              Y_Mean
              Y_Sigma
              DC_EP
              TCD_TC
              Cmin
              Cmax
              EP_LP
Test mode:    split 50% train, remainder test
```

```
=== Classifier model (full training set) ===

IB1 instance-based classifier
using 9 nearest neighbour(s) for classification

Time taken to build model: 0 seconds

=== Predictions on test split ===

 inst#,     actual, predicted, error, probability distribution
     1        1:Y        1:Y          *0.777  0.223
     2        2:N        2:N           0.223 *0.777
     3        1:Y        1:Y          *0.777  0.223
     4        1:Y        1:Y          *0.666  0.334
     5        2:N        1:Y      +   *0.666  0.334
     6        2:N        2:N           0.112 *0.888
     7        2:N        1:Y      +   *0.777  0.223
     8        1:Y        1:Y          *0.888  0.112
     9        1:Y        2:N      +    0.223 *0.777
    10        1:Y        1:Y          *0.666  0.334
    11        1:Y        2:N      +    0.334 *0.666
    12        1:Y        1:Y          *0.888  0.112
    13        1:Y        1:Y          *0.888  0.112
    14        1:Y        1:Y          *0.888  0.112
    15        1:Y        1:Y          *0.998  0.002
    16        2:N        2:N           0.223 *0.777
    17        1:Y        1:Y          *0.777  0.223
    18        2:N        2:N           0.002 *0.998
    19        1:Y        1:Y          *0.998  0.002
    20        1:Y        1:Y          *0.666  0.334
    21        1:Y        1:Y          *0.888  0.112
    22        2:N        2:N           0.223 *0.777
    23        1:Y        2:N      +    0.334 *0.666
    24        1:Y        1:Y          *0.888  0.112
    25        1:Y        1:Y          *0.777  0.223
    26        1:Y        1:Y          *0.998  0.002
```

```
27       2:N       2:N          0.002 *0.998
28       1:Y       1:Y        *0.888  0.112
29       1:Y       1:Y        *0.666  0.334
30       2:N       2:N          0.223 *0.777
31       2:N       2:N          0.223 *0.777
32       1:Y       1:Y        *0.888  0.112
33       2:N       1:Y    +   *0.555  0.445
34       2:N       2:N          0.223 *0.777
35       1:Y       1:Y        *0.888  0.112
36       2:N       2:N          0.223 *0.777
37       1:Y       2:N    +     0.334 *0.666
38       1:Y       1:Y        *0.666  0.334
39       1:Y       1:Y        *0.666  0.334
40       2:N       2:N          0.112 *0.888
41       1:Y       2:N    +     0.445 *0.555
42       2:N       2:N          0.334 *0.666
43       1:Y       1:Y        *0.998  0.002
44       1:Y       1:Y        *0.888  0.112
45       2:N       2:N          0.112 *0.888
46       2:N       2:N          0.223 *0.777
47       2:N       1:Y    +   *0.777  0.223
48       2:N       2:N          0.002 *0.998
49       1:Y       1:Y        *0.998  0.002
50       1:Y       1:Y        *0.888  0.112
51       1:Y       1:Y        *0.998  0.002
52       2:N       2:N          0.112 *0.888
53       2:N       1:Y    +   *0.555  0.445
54       1:Y       1:Y        *0.888  0.112
55       1:Y       1:Y        *0.998  0.002
56       1:Y       1:Y        *0.888  0.112
57       2:N       1:Y    +   *0.666  0.334
58       2:N       1:Y    +   *0.555  0.445
59       1:Y       1:Y        *0.777  0.223
60       1:Y       1:Y        *0.888  0.112
61       1:Y       1:Y        *0.888  0.112
62       1:Y       1:Y        *0.998  0.002
```

```
    63        1:Y        1:Y          *0.555  0.445
    64        1:Y        1:Y          *0.666  0.334
    65        1:Y        1:Y          *0.777  0.223
    66        1:Y        1:Y          *0.777  0.223
    67        2:N        1:Y      +   *0.777  0.223
    68        1:Y        1:Y          *0.888  0.112

=== Evaluation on test split ===
=== Summary ===

Correctly Classified Instances          55               80.8824 %
Incorrectly Classified Instances        13               19.1176 %
Kappa statistic                          0.5692
K&B Relative Info Score               3225.57   %
K&B  Information  Score                          28.4267  bits         0.418
bits/instance
Class  complexity  |  order  0                    64.3284  bits         0.946
bits/instance
Class  complexity  |  scheme                     35.7135  bits         0.5252
bits/instance
Complexity  improvement       (Sf)              28.6149  bits         0.4208
bits/instance
Mean absolute error                      0.2557
Root mean squared error                  0.342
Relative absolute error                 57.9587 %
Root relative squared error             71.1383 %
Total Number of Instances               68

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   Class
  0.886     0.333      0.83      0.886     0.857    Y
  0.667     0.114      0.762     0.667     0.711    N
```

```
=== Confusion Matrix ===

  a  b   <-- classified as
 39  5 |  a = Y
  8 16 |  b = N
```

```
Scheme:       weka.classifiers.lazy.IBk -K 9 -W 0
Relation:     Image_Features
Instances:    136
Attributes:   23
              File_Name
              Malignant
              Lesion_Nr_Points
              Edge_Nr_Points
              Dir_Changes
              Rays_Sigma
              C_Total
              C_Mean
              C_Sigma
              CD_Total
              CD_Mean
              CD_Sigma
              X_Total
              X_Mean
              X_Sigma
              Y_Total
              Y_Mean
              Y_Sigma
              DC_EP
              TCD_TC
              Cmin
              Cmax
              EP_LP
Test mode:    split 60% train, remainder test

=== Classifier model (full training set) ===

IB1 instance-based classifier
using 9 nearest neighbour(s) for classification


Time taken to build model: 0 seconds
```

```
=== Predictions on test split ===

 inst#,     actual, predicted, error, probability distribution
     1        1:Y        1:Y         *0.888  0.112
     2        1:Y        1:Y         *0.999  0.001
     3        2:N        2:N          0.112 *0.888
     4        1:Y        1:Y         *0.777  0.223
     5        2:N        2:N          0.223 *0.777
     6        1:Y        1:Y         *0.999  0.001
     7        1:Y        1:Y         *0.555  0.445
     8        1:Y        1:Y         *0.888  0.112
     9        2:N        2:N          0.223 *0.777
    10        1:Y        1:Y         *0.555  0.445
    11        1:Y        1:Y         *0.888  0.112
    12        1:Y        1:Y         *0.777  0.223
    13        1:Y        1:Y         *0.999  0.001
    14        2:N        2:N          0.112 *0.888
    15        1:Y        1:Y         *0.888  0.112
    16        1:Y        1:Y         *0.888  0.112
    17        2:N        2:N          0.112 *0.888
    18        2:N        2:N          0.112 *0.888
    19        1:Y        1:Y         *0.999  0.001
    20        2:N        1:Y      +  *0.666  0.334
    21        2:N        2:N          0.223 *0.777
    22        1:Y        1:Y         *0.888  0.112
    23        2:N        2:N          0.445 *0.555
    24        1:Y        2:N      +   0.445 *0.555
    25        1:Y        1:Y         *0.666  0.334
    26        1:Y        1:Y         *0.666  0.334
    27        2:N        2:N          0.001 *0.999
    28        1:Y        1:Y         *0.666  0.334
    29        2:N        2:N          0.223 *0.777
    30        1:Y        1:Y         *0.888  0.112
    31        1:Y        1:Y         *0.888  0.112
    32        2:N        2:N          0.112 *0.888
    33        2:N        2:N          0.001 *0.999
```

```
34       2:N       1:Y       +  *0.777  0.223
35       2:N       2:N          0.223 *0.777
36       1:Y       1:Y          *0.999  0.001
37       1:Y       1:Y          *0.888  0.112
38       1:Y       1:Y          *0.888  0.112
39       2:N       2:N          0.223 *0.777
40       2:N       1:Y       +  *0.555  0.445
41       1:Y       1:Y          *0.777  0.223
42       1:Y       1:Y          *0.999  0.001
43       1:Y       1:Y          *0.777  0.223
44       2:N       1:Y       +  *0.555  0.445
45       2:N       2:N          0.445 *0.555
46       1:Y       1:Y          *0.777  0.223
47       1:Y       1:Y          *0.888  0.112
48       1:Y       1:Y          *0.888  0.112
49       1:Y       1:Y          *0.999  0.001
50       1:Y       1:Y          *0.555  0.445
51       1:Y       1:Y          *0.666  0.334
52       1:Y       1:Y          *0.666  0.334
53       1:Y       1:Y          *0.777  0.223
54       2:N       1:Y       +  *0.666  0.334
55       1:Y       1:Y          *0.888  0.112
```

```
=== Evaluation on test split ===
=== Summary ===

Correctly Classified Instances          49               89.0909 %
Incorrectly Classified Instances         6               10.9091 %
Kappa statistic                          0.7537
K&B Relative Info Score               3194.1903 %
K&B Information Score                   28.197  bits      0.5127
bits/instance
Class complexity | order 0              52.7207 bits      0.9586
bits/instance
Class complexity | scheme               23.4863 bits      0.427
bits/instance
Complexity improvement     (Sf)         29.2344 bits      0.5315
bits/instance
Mean absolute error                      0.225
Root mean squared error                  0.295
Relative absolute error                 50.4723 %
Root relative squared error             60.8087 %
Total Number of Instances               55

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   Class
  0.971     0.25       0.872     0.971     0.919    Y
  0.75      0.029      0.938     0.75      0.833    N

=== Confusion Matrix ===

  a  b   <-- classified as
 34  1 |  a = Y
  5 15 |  b = N
```

```
Scheme:       weka.classifiers.lazy.IBk -K 9 -W 0
Relation:     Image_Features
Instances:    136
Attributes:   23
              File_Name
              Malignant
              Lesion_Nr_Points
              Edge_Nr_Points
              Dir_Changes
              Rays_Sigma
              C_Total
              C_Mean
              C_Sigma
              CD_Total
              CD_Mean
              CD_Sigma
              X_Total
              X_Mean
              X_Sigma
              Y_Total
              Y_Mean
              Y_Sigma
              DC_EP
              TCD_TC
              Cmin
              Cmax
              EP_LP
Test mode:    split 70% train, remainder test

=== Classifier model (full training set) ===

IB1 instance-based classifier
using 9 nearest neighbour(s) for classification


Time taken to build model: 0 seconds
```

```
=== Predictions on test split ===

 inst#,     actual, predicted, error, probability distribution
      1        1:Y        1:Y         *0.888  0.112
      2        1:Y        1:Y         *0.888  0.112
      3        2:N        2:N          0.112 *0.888
      4        2:N        2:N          0.112 *0.888
      5        1:Y        1:Y         *0.999  0.001
      6        2:N        1:Y      +  *0.666  0.334
      7        2:N        2:N          0.223 *0.777
      8        1:Y        1:Y         *0.888  0.112
      9        2:N        1:Y      +  *0.555  0.445
     10        1:Y        2:N      +   0.445 *0.555
     11        1:Y        1:Y         *0.666  0.334
     12        1:Y        1:Y         *0.666  0.334
     13        2:N        2:N          0.001 *0.999
     14        1:Y        1:Y         *0.666  0.334
     15        2:N        2:N          0.112 *0.888
     16        1:Y        1:Y         *0.888  0.112
     17        1:Y        1:Y         *0.888  0.112
     18        2:N        2:N          0.112 *0.888
     19        2:N        2:N          0.001 *0.999
     20        2:N        1:Y      +  *0.888  0.112
     21        2:N        2:N          0.334 *0.666
     22        1:Y        1:Y         *0.999  0.001
     23        1:Y        1:Y         *0.888  0.112
     24        1:Y        1:Y         *0.888  0.112
     25        2:N        2:N          0.334 *0.666
     26        2:N        1:Y      +  *0.666  0.334
     27        1:Y        1:Y         *0.777  0.223
     28        1:Y        1:Y         *0.999  0.001
     29        1:Y        1:Y         *0.777  0.223
     30        2:N        1:Y      +  *0.555  0.445
     31        2:N        2:N          0.445 *0.555
     32        1:Y        1:Y         *0.888  0.112
     33        1:Y        1:Y         *0.888  0.112
```

```
    34        1:Y        1:Y          *0.888  0.112
    35        1:Y        1:Y          *0.999  0.001
    36        1:Y        1:Y          *0.555  0.445
    37        1:Y        1:Y          *0.666  0.334
    38        1:Y        1:Y          *0.666  0.334
    39        1:Y        1:Y          *0.888  0.112
    40        2:N        1:Y      +   *0.666  0.334
    41        1:Y        1:Y          *0.999  0.001

=== Evaluation on test split ===
=== Summary ===

Correctly Classified Instances          34               82.9268 %
Incorrectly Classified Instances         7               17.0732 %
Kappa statistic                          0.6199
K&B Relative Info Score               2269.2325 %
K&B Information Score                   19.9699 bits      0.4871
bits/instance
Class complexity | order 0              40.6822 bits      0.9922
bits/instance
Class complexity | scheme               20.8879 bits      0.5095
bits/instance
Complexity improvement     (Sf)         19.7942 bits      0.4828
bits/instance
Mean absolute error                      0.2472
Root mean squared error                  0.3334
Relative absolute error                 54.2267 %
Root relative squared error             67.185  %
Total Number of Instances               41

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   Class
  0.96      0.375      0.8       0.96      0.873    Y
  0.625     0.04       0.909     0.625     0.741    N
```

```
=== Confusion Matrix ===

  a  b   <-- classified as
 24  1 |  a = Y
  6 10 |  b = N
```

```
Scheme:       weka.classifiers.lazy.IBk -K 9 -W 0
Relation:     Image_Features
Instances:    136
Attributes:   23
              File_Name
              Malignant
              Lesion_Nr_Points
              Edge_Nr_Points
              Dir_Changes
              Rays_Sigma
              C_Total
              C_Mean
              C_Sigma
              CD_Total
              CD_Mean
              CD_Sigma
              X_Total
              X_Mean
              X_Sigma
              Y_Total
              Y_Mean
              Y_Sigma
              DC_EP
              TCD_TC
              Cmin
              Cmax
              EP_LP
Test mode:    split 80% train, remainder test

=== Classifier model (full training set) ===

IB1 instance-based classifier
using 9 nearest neighbour(s) for classification


Time taken to build model: 0 seconds
```

```
=== Predictions on test split ===

 inst#,     actual, predicted, error, probability distribution
     1        1:Y        1:Y          *0.555  0.445
     2        2:N        2:N           0.001 *0.999
     3        1:Y        1:Y          *0.888  0.112
     4        1:Y        1:Y          *0.888  0.112
     5        2:N        2:N           0.223 *0.777
     6        2:N        2:N           0.001 *0.999
     7        2:N        1:Y      +   *0.777  0.223
     8        2:N        2:N           0.334 *0.666
     9        1:Y        1:Y          *0.999  0.001
    10        1:Y        1:Y          *0.777  0.223
    11        1:Y        1:Y          *0.888  0.112
    12        2:N        2:N           0.334 *0.666
    13        2:N        1:Y      +   *0.555  0.445
    14        1:Y        1:Y          *0.777  0.223
    15        1:Y        1:Y          *0.999  0.001
    16        1:Y        1:Y          *0.777  0.223
    17        2:N        2:N           0.223 *0.777
    18        2:N        2:N           0.334 *0.666
    19        1:Y        1:Y          *0.888  0.112
    20        1:Y        1:Y          *0.888  0.112
    21        1:Y        1:Y          *0.888  0.112
    22        1:Y        1:Y          *0.999  0.001
    23        1:Y        1:Y          *0.555  0.445
    24        1:Y        1:Y          *0.666  0.334
    25        1:Y        1:Y          *0.666  0.334
    26        1:Y        1:Y          *0.888  0.112
    27        2:N        1:Y      +   *0.666  0.334
    28        1:Y        1:Y          *0.999  0.001
```

```
=== Evaluation on test split ===
=== Summary ===

Correctly Classified Instances          25               89.2857 %
Incorrectly Classified Instances         3               10.7143 %
Kappa statistic                          0.75
K&B Relative Info Score               1526.7608 %
K&B Information Score                   13.7774 bits      0.492
bits/instance
Class complexity | order 0              26.4665 bits      0.9452
bits/instance
Class complexity | scheme               12.57   bits      0.4489
bits/instance
Complexity improvement     (Sf)         13.8966 bits      0.4963
bits/instance
Mean absolute error                      0.2307
Root mean squared error                  0.3073
Relative absolute error                 51.4924 %
Root relative squared error             63.9253 %
Total Number of Instances               28

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   Class
  1         0.3        0.857     1         0.923    Y
  0.7       0          1         0.7       0.824    N

=== Confusion Matrix ===

  a  b   <-- classified as
 18  0 |  a = Y
  3  7 |  b = N
```

```
Scheme:       weka.classifiers.lazy.IBk -K 9 -W 0
Relation:     Image_Features
Instances:    136
Attributes:   23
              File_Name
              Malignant
              Lesion_Nr_Points
              Edge_Nr_Points
              Dir_Changes
              Rays_Sigma
              C_Total
              C_Mean
              C_Sigma
              CD_Total
              CD_Mean
              CD_Sigma
              X_Total
              X_Mean
              X_Sigma
              Y_Total
              Y_Mean
              Y_Sigma
              DC_EP
              TCD_TC
              Cmin
              Cmax
              EP_LP
Test mode:    10-fold cross-validation

=== Classifier model (full training set) ===

IB1 instance-based classifier
using 9 nearest neighbour(s) for classification


Time taken to build model: 0 seconds
```

```
=== Predictions on test data ===

 inst#,     actual, predicted, error, probability distribution
     1        2:N        2:N          0.001 *0.999
     2        2:N        2:N          0.445 *0.555
     3        2:N        2:N          0.112 *0.888
     4        2:N        1:Y      +  *0.666  0.334
     5        2:N        1:Y      +  *0.555  0.445
     6        1:Y        1:Y         *0.888  0.112
     7        1:Y        1:Y         *0.555  0.445
     8        1:Y        1:Y         *0.666  0.334
     9        1:Y        1:Y         *0.777  0.223
    10        1:Y        1:Y         *0.555  0.445
    11        1:Y        1:Y         *0.999  0.001
    12        1:Y        1:Y         *0.777  0.223
    13        1:Y        1:Y         *0.888  0.112
    14        1:Y        1:Y         *0.555  0.445
     1        2:N        2:N          0.112 *0.888
     2        2:N        2:N          0.223 *0.777
     3        2:N        1:Y      +  *0.777  0.223
     4        2:N        2:N          0.334 *0.666
     5        2:N        2:N          0.112 *0.888
     6        1:Y        1:Y         *0.999  0.001
     7        1:Y        2:N      +   0.445 *0.555
     8        1:Y        1:Y         *0.666  0.334
     9        1:Y        1:Y         *0.777  0.223
    10        1:Y        1:Y         *0.999  0.001
    11        1:Y        1:Y         *0.777  0.223
    12        1:Y        1:Y         *0.999  0.001
    13        1:Y        1:Y         *0.888  0.112
    14        1:Y        1:Y         *0.777  0.223
     1        2:N        2:N          0.334 *0.666
     2        2:N        1:Y      +  *0.666  0.334
     3        2:N        2:N          0.001 *0.999
     4        2:N        1:Y      +  *0.666  0.334
```

```
 5        2:N        2:N          0.223 *0.777
 6        1:Y        1:Y        *0.999  0.001
 7        1:Y        1:Y        *0.999  0.001
 8        1:Y        1:Y        *0.666  0.334
 9        1:Y        1:Y        *0.999  0.001
10        1:Y        1:Y        *0.888  0.112
11        1:Y        1:Y        *0.999  0.001
12        1:Y        1:Y        *0.999  0.001
13        1:Y        1:Y        *0.999  0.001
14        1:Y        1:Y        *0.777  0.223
 1        2:N        1:Y      +  *0.777  0.223
 2        2:N        2:N          0.445 *0.555
 3        2:N        1:Y      +  *0.666  0.334
 4        2:N        2:N          0.112 *0.888
 5        2:N        1:Y      +  *0.666  0.334
 6        1:Y        1:Y        *0.888  0.112
 7        1:Y        1:Y        *0.999  0.001
 8        1:Y        1:Y        *0.999  0.001
 9        1:Y        1:Y        *0.777  0.223
10        1:Y        1:Y        *0.999  0.001
11        1:Y        1:Y        *0.777  0.223
12        1:Y        1:Y        *0.999  0.001
13        1:Y        1:Y        *0.777  0.223
14        1:Y        1:Y        *0.888  0.112
 1        2:N        2:N          0.001 *0.999
 2        2:N        2:N          0.112 *0.888
 3        2:N        2:N          0.112 *0.888
 4        2:N        2:N          0.001 *0.999
 5        1:Y        1:Y        *0.888  0.112
 6        1:Y        1:Y        *0.888  0.112
 7        1:Y        1:Y        *0.999  0.001
 8        1:Y        1:Y        *0.888  0.112
 9        1:Y        1:Y        *0.777  0.223
10        1:Y        2:N      +   0.445 *0.555
11        1:Y        2:N      +   0.223 *0.777
12        1:Y        1:Y        *0.999  0.001
```

```
 13        1:Y        1:Y          *0.777  0.223
 14        1:Y        1:Y          *0.888  0.112
  1        2:N        2:N           0.001 *0.999
  2        2:N        2:N           0.223 *0.777
  3        2:N        2:N           0.334 *0.666
  4        2:N        2:N           0.334 *0.666
  5        1:Y        1:Y          *0.999  0.001
  6        1:Y        1:Y          *0.888  0.112
  7        1:Y        2:N      +    0.334 *0.666
  8        1:Y        1:Y          *0.888  0.112
  9        1:Y        1:Y          *0.999  0.001
 10        1:Y        1:Y          *0.777  0.223
 11        1:Y        1:Y          *0.555  0.445
 12        1:Y        1:Y          *0.888  0.112
 13        1:Y        1:Y          *0.888  0.112
 14        1:Y        1:Y          *0.999  0.001
  1        2:N        1:Y      +   *0.999  0.001
  2        2:N        1:Y      +   *0.888  0.112
  3        2:N        2:N           0.112 *0.888
  4        2:N        2:N           0.001 *0.999
  5        1:Y        1:Y          *0.999  0.001
  6        1:Y        1:Y          *0.999  0.001
  7        1:Y        1:Y          *0.777  0.223
  8        1:Y        2:N      +    0.112 *0.888
  9        1:Y        1:Y          *0.999  0.001
 10        1:Y        1:Y          *0.666  0.334
 11        1:Y        1:Y          *0.666  0.334
 12        1:Y        1:Y          *0.999  0.001
 13        1:Y        1:Y          *0.999  0.001
  1        2:N        2:N           0.001 *0.999
  2        2:N        1:Y      +   *0.888  0.112
  3        2:N        2:N           0.112 *0.888
  4        2:N        1:Y      +   *0.777  0.223
  5        1:Y        1:Y          *0.999  0.001
  6        1:Y        1:Y          *0.777  0.223
  7        1:Y        1:Y          *0.666  0.334
```

```
 8        1:Y        1:Y          *0.555  0.445
 9        1:Y        1:Y          *0.555  0.445
10        1:Y        2:N      +    0.334 *0.666
11        1:Y        1:Y          *0.777  0.223
12        1:Y        1:Y          *0.777  0.223
13        1:Y        1:Y          *0.888  0.112
 1        2:N        2:N           0.334 *0.666
 2        2:N        2:N           0.223 *0.777
 3        2:N        2:N           0.001 *0.999
 4        2:N        2:N           0.223 *0.777
 5        1:Y        1:Y          *0.999  0.001
 6        1:Y        1:Y          *0.888  0.112
 7        1:Y        1:Y          *0.777  0.223
 8        1:Y        2:N      +    0.445 *0.555
 9        1:Y        1:Y          *0.999  0.001
10        1:Y        1:Y          *0.999  0.001
11        1:Y        1:Y          *0.666  0.334
12        1:Y        1:Y          *0.777  0.223
13        1:Y        1:Y          *0.888  0.112
 1        2:N        1:Y      +   *0.777  0.223
 2        2:N        2:N           0.001 *0.999
 3        2:N        2:N           0.223 *0.777
 4        2:N        2:N           0.334 *0.666
 5        1:Y        1:Y          *0.999  0.001
 6        1:Y        1:Y          *0.999  0.001
 7        1:Y        1:Y          *0.888  0.112
 8        1:Y        1:Y          *0.999  0.001
 9        1:Y        1:Y          *0.999  0.001
10        1:Y        1:Y          *0.999  0.001
11        1:Y        1:Y          *0.888  0.112
12        1:Y        1:Y          *0.888  0.112
13        1:Y        1:Y          *0.999  0.001
```

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         116               85.2941 %
Incorrectly Classified Instances        20               14.7059 %
Kappa statistic                          0.6516
K&B Relative Info Score               6670.6489 %
K&B Information Score                   60.8943 bits      0.4478
bits/instance
Class complexity | order 0             123.6058 bits      0.9089
bits/instance
Class complexity | scheme               74.1642 bits      0.5453
bits/instance
Complexity improvement     (Sf)         49.4416 bits      0.3635
bits/instance
Mean absolute error                      0.2268
Root mean squared error                  0.3354
Relative absolute error                 51.6766 %
Root relative squared error             71.654  %
Total Number of Instances              136

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   Class
  0.924     0.295      0.867     0.924     0.895    Y
  0.705     0.076      0.816     0.705     0.756    N

=== Confusion Matrix ===

  a  b   <-- classified as
 85  7 |  a = Y
 13 31 |  b = N
```

```
Scheme:          weka.classifiers.functions.SMO -C 1.0 -E 1.0 -G 0.01 -A
250007 -L 0.0010 -P 1.0E-12 -N 0 -V -1 -W 1
Relation:     Image_Features
Instances:    136
Attributes:   23
              File_Name
              Malignant
              Lesion_Nr_Points
              Edge_Nr_Points
              Dir_Changes
              Rays_Sigma
              C_Total
              C_Mean
              C_Sigma
              CD_Total
              CD_Mean
              CD_Sigma
              X_Total
              X_Mean
              X_Sigma
              Y_Total
              Y_Mean
              Y_Sigma
              DC_EP
              TCD_TC
              Cmin
              Cmax
              EP_LP
Test mode:    split 50% train, remainder test
```

```
=== Classifier model (full training set) ===

SMO

Classifier for classes: Y, N

BinarySMO

Machine linear: showing attribute weights, not support vectors.

        -0.1488 * (normalized) File_Name=Basal_img0019a_Combi_T1.png
 +      -1      * (normalized) File_Name=Basalioma_01_Combi_T1.png
 +      -0.0996 * (normalized) File_Name=Basalioma_02_Combi_T1.png
 +      -1      * (normalized) File_Name=Basalioma_03_Combi_T4.png
 +      -0.7022 * (normalized) File_Name=basalioma_a09f8_Combi_T1.png
 +      -0.0038 * (normalized) File_Name=Carcinoma_basal_02_Combi_T8.png
 +      -0.2634 * (normalized) File_Name=1287melanoma2_Combi_T9.png
 +      -0.2812 * (normalized) File_Name=image_a_Combi_T1.png
 +      -0.0308 * (normalized) File_Name=malig2_Combi_T1.png
 +      -0.243  * (normalized) File_Name=malignant_melanoma_2_Combi_T6.png
 +      -0.4506 * (normalized) File_Name=Melanoma_04_Combi_T6.png
 +      -0.4009 * (normalized) File_Name=Melanoma_005a_Combi_T3.png
 +      -0.6554 * (normalized) File_Name=Melanoma_006_Combi_T1.png
 +      -0.7613 * (normalized) File_Name=melanoma_007_Combi_T10.png
 +      -1      * (normalized) File_Name=melanoma_01_Combi_T6.png
 +      -0.5596 * (normalized) File_Name=melanoma_012_Combi_T1.png
 +      -0.4714 * (normalized) File_Name=melanoma_014_Combi_T1.png
 +      -0.0176 * (normalized) File_Name=Melanoma_016_Combi_T1.png
 +      -0.0856 * (normalized) File_Name=melanoma_018_Combi_T1.png
 +      -1      * (normalized) File_Name=Melanoma_02_Combi_T10.png
 +      -0.1285 * (normalized) File_Name=melanoma_a09f2_Combi_T1.png
 +      -0.2052 * (normalized) File_Name=melanoma_abdc_01_Combi_T12.png
 +      -0.1663 * (normalized) File_Name=melanoma_abdc_02_Combi_T1.png
 +      -0.3688 * (normalized) File_Name=melanoma_abdc_03_Combi_T2.png
 +      -0.3157 * (normalized) File_Name=Melanoma_img0002_Combi_T6.png
 +      -0.3478 * (normalized) File_Name=Melanoma_img0004_Combi_T1.png
```

```
 +      -0.2499 * (normalized) File_Name=Melanoma_img0005a_Combi_T1.png
 +      -0.5686 * (normalized) File_Name=Melanoma_img0033_Combi_T1.png
 +      -1      * (normalized) File_Name=Melanoma_img0048_Combi_T1.png
 +      -0.109  * (normalized) File_Name=Melanoma_img0056_Combi_T1.png
 +      -0.844  * (normalized) File_Name=Melanoma_img0081_Combi_T1.png
 +      -0.0619 * (normalized) File_Name=Melanoma_img0085a_Combi_T1.png
 +      -0.0239 * (normalized) File_Name=Melanoma_img0090_Combi_T5.png
 +      -0.2876 * (normalized) File_Name=Melanoma_img0090a_Combi_T1.png
 +      -0.1329 * (normalized) File_Name=Melanoma_img0092_Combi_T1.png
 +      -0.3605 * (normalized) File_Name=Melanoma_img0095a_Combi_T5.png
 +      -0.0661 * (normalized) File_Name=Melanoma_img0097_Combi_T1.png
 +      -0.0703 * (normalized) File_Name=Melanoma_img0105_Combi_T1.png
 +      -1      * (normalized) File_Name=melanoma_mal_Combi_T4.png
 +      -0.7867 * (normalized) File_Name=melanoma_nodule_Combi_T1.png
 +      -0.7253 * (normalized) File_Name=melanoma_palpabile_Combi_T1.png
 +      -0.2139 * (normalized) File_Name=melanoma-1_Combi_T4.png
 +      -0.7923 * (normalized) File_Name=melanoma-2_Combi_T10.png
 +      -0.1614 * (normalized) File_Name=melanoma4_Combi_T6.png
 +      -1      * (normalized) File_Name=melanoma8_Combi_T10.png
 +      -0.1177 * (normalized) File_Name=melanoma-fig3_Combi_T1.pn
 +      -0.7512 * (normalized) File_Name=melanoma-fig4_Combi_T3.pn
 +      -0.1648 * (normalized) File_Name=no_5thn_Combi_T1.png
 +      -1      * (normalized) File_Name=Escamosas_img0046_Combi_T1.png
 +       0.1381 * (normalized) File_Name=Atypical_mole_001_Combi_T6.png
 +       0.6393 * (normalized) File_Name=Atypical_mole_002_Combi_T4.png
 +       0.9589 * (normalized) File_Name=keratosis1_Combi_T1.png
 +       0.66   * (normalized) File_Name=Queratose_img0088_Combi_T4.png
 +       1      * (normalized) File_Name=Queratose_img0093_Combi_T1.png
 +       1      * (normalized) File_Name=Queratose_img0094_Combi_T1.png
 +       1      * (normalized) File_Name=Queratose_img0095_Combi_T12.png
 +                              1                          *     (normalized)
File_Name=seborrheic_keratosis_02_Combi_T1.png
 +                              1                          *     (normalized)
File_Name=seborrheic_keratosis_03_Combi_T4.png
 +       0.5909 * (normalized) File_Name=Atypical nevus_01_Combi_T1.png
 +       1      * (normalized) File_Name=Benign nevus_01_Combi_T4.png
```

```
+       0.2781 * (normalized) File_Name=Benign nevus_02_Combi_T1.png
+       1      * (normalized) File_Name=Benign nevus_03_Combi_T11.png
+       0.7963 * (normalized) File_Name=nevi4_Combi_T1.png
+       0.6617 * (normalized) File_Name=nevi4a_Combi_T1.png
+       0.5308 * (normalized) File_Name=nevo_03_Combi_T3.png
+       0.2782 * (normalized) File_Name=nevo_04_Combi_T1.png
+       0.0023 * (normalized) File_Name=nevo_congenito_Combi_T1.png
+       0.3794 * (normalized) File_Name=nevo_img0030_Combi_T1.png
+       1      * (normalized) File_Name=Nevo_img0031_Combi_T10.png
+       1      * (normalized) File_Name=nevo_img0084_Combi_T2.png
+       0.6253 * (normalized) File_Name=nevo_lentigginoso_Combi_T1.png
+       0.722  * (normalized) File_Name=nevoa3_small_Combi_T1.png
+       0.2723 * (normalized) File_Name=nevoc2_small_Combi_T1.png
+       0.1918 * (normalized) File_Name=nevodis_small_Combi_T0.png
+       0.3391 * (normalized) File_Name=nevosp1_small_Combi_T1.png
+       0.7253 * (normalized) File_Name=nevosu_small1_Combi_T3.png
+       0.1489 * (normalized) File_Name=nevou5_small_Combi_T1.png
+       0.2467 * (normalized) File_Name=Nevus_003_Combi_T1.png
+       0.0787 * (normalized) File_Name=Nevus_img0085_Combi_T4.png
+       0.0978 * (normalized) File_Name=nvmelcomp_a_Combi_T1.png
+       0.0051 * (normalized) File_Name=nvmelintra0_Combi_T1.png
+       0.7159 * (normalized) File_Name=nvmelpeq_Combi_T1.png
+       1      * (normalized) File_Name=Naevi_melanocytic3a_Combi_T1.png
+       0.6561 * (normalized) File_Name=Naevi_melanocytic3b_Combi_T1.png
+       0.0789 * (normalized) File_Name=Carolina_01
+       0.0993 * (normalized) File_Name=JCV_6mm
+       0.2781 * (normalized) File_Name=Maria_01
+       0.2534 * (normalized) Lesion_Nr_Points
+      -0.6929 * (normalized) Edge_Nr_Points
+      -0.7738 * (normalized) Dir_Changes
+      -0.7596 * (normalized) Rays_Sigma
+      -0.0789 * (normalized) C_Total
+       0.062  * (normalized) C_Mean
+      -0.5696 * (normalized) C_Sigma
+      -0.142  * (normalized) CD_Total
+      -0.2662 * (normalized) CD_Mean
```

```
 +      -0.4975 * (normalized) CD_Sigma
 +      -0.0114 * (normalized) X_Total
 +      -0.5448 * (normalized) X_Mean
 +      -0.8227 * (normalized) X_Sigma
 +      -0.0573 * (normalized) Y_Total
 +      -0.5068 * (normalized) Y_Mean
 +      -0.0144 * (normalized) Y_Sigma
 +      -0.6413 * (normalized) DC_EP
 +      -0.114  * (normalized) TCD_TC
 +       0.3124 * (normalized) Cmin
 +      -0.1937 * (normalized) Cmax
 +       0.0743 * (normalized) EP_LP
 +       1.5067

Number of kernel evaluations: 5402 (86.918% cached)



Time taken to build model: 0.09 seconds
```

```
=== Predictions on test split ===

 inst#,     actual, predicted, error, probability distribution
      1        2:N        2:N          0     *1
      2        1:Y        1:Y         *1      0
      3        1:Y        1:Y         *1      0
      4        2:N        2:N          0     *1
      5        2:N        2:N          0     *1
      6        1:Y        1:Y         *1      0
      7        1:Y        1:Y         *1      0
      8        1:Y        1:Y         *1      0
      9        1:Y        1:Y         *1      0
     10        1:Y        1:Y         *1      0
     11        2:N        1:Y      +  *1      0
     12        2:N        2:N          0     *1
     13        1:Y        1:Y         *1      0
     14        1:Y        1:Y         *1      0
     15        2:N        1:Y      +  *1      0
     16        1:Y        1:Y         *1      0
     17        1:Y        1:Y         *1      0
     18        1:Y        2:N      +   0     *1
     19        2:N        1:Y      +  *1      0
     20        2:N        1:Y      +  *1      0
     21        1:Y        1:Y         *1      0
     22        1:Y        1:Y         *1      0
     23        2:N        2:N          0     *1
     24        2:N        2:N          0     *1
     25        1:Y        1:Y         *1      0
     26        1:Y        1:Y         *1      0
     27        2:N        2:N          0     *1
     28        1:Y        1:Y         *1      0
     29        1:Y        1:Y         *1      0
     30        2:N        2:N          0     *1
     31        1:Y        1:Y         *1      0
     32        1:Y        1:Y         *1      0
     33        1:Y        1:Y         *1      0
```

34	1:Y	1:Y		*1	0
35	1:Y	1:Y		*1	0
36	1:Y	1:Y		*1	0
37	2:N	2:N		0	*1
38	2:N	1:Y	+	*1	0
39	1:Y	1:Y		*1	0
40	2:N	2:N		0	*1
41	1:Y	1:Y		*1	0
42	1:Y	1:Y		*1	0
43	1:Y	2:N	+	0	*1
44	2:N	2:N		0	*1
45	1:Y	1:Y		*1	0
46	1:Y	1:Y		*1	0
47	1:Y	1:Y		*1	0
48	2:N	1:Y	+	*1	0
49	1:Y	1:Y		*1	0
50	1:Y	1:Y		*1	0
51	2:N	2:N		0	*1
52	1:Y	1:Y		*1	0
53	2:N	1:Y	+	*1	0
54	1:Y	1:Y		*1	0
55	2:N	1:Y	+	*1	0
56	1:Y	1:Y		*1	0
57	2:N	1:Y	+	*1	0
58	1:Y	1:Y		*1	0
59	1:Y	1:Y		*1	0
60	2:N	2:N		0	*1
61	1:Y	1:Y		*1	0
62	2:N	2:N		0	*1
63	2:N	2:N		0	*1
64	1:Y	1:Y		*1	0
65	1:Y	1:Y		*1	0
66	1:Y	1:Y		*1	0
67	1:Y	2:N	+	0	*1
68	1:Y	1:Y		*1	0

```
=== Evaluation on test split ===
=== Summary ===

Correctly Classified Instances          56               82.3529 %
Incorrectly Classified Instances        12               17.6471 %
Kappa statistic                          0.5904
K&B Relative Info Score               4233.5543 %
K&B Information Score                           37.3099 bits        0.5487
bits/instance
Class complexity | order 0                      64.3284 bits         0.946
bits/instance
Class complexity | scheme                 12888         bits      189.5294
bits/instance
Complexity improvement         (Sf)         -12823.6716 bits     -188.5834
bits/instance
Mean absolute error                      0.1765
Root mean squared error                  0.4201
Relative absolute error                 40      %
Root relative squared error             87.3704 %
Total Number of Instances               68

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   Class
  0.932     0.375      0.82      0.932     0.872    Y
  0.625     0.068      0.833     0.625     0.714    N

=== Confusion Matrix ===

  a  b   <-- classified as
 41  3 |  a = Y
  9 15 |  b = N
```

```
Scheme:          weka.classifiers.functions.SMO -C 1.0 -E 1.0 -G 0.01 -A
250007 -L 0.0010 -P 1.0E-12 -N 0 -V -1 -W 1
Relation:     Image_Features
Instances:    136
Attributes:   23
              File_Name
              Malignant
              Lesion_Nr_Points
              Edge_Nr_Points
              Dir_Changes
              Rays_Sigma
              C_Total
              C_Mean
              C_Sigma
              CD_Total
              CD_Mean
              CD_Sigma
              X_Total
              X_Mean
              X_Sigma
              Y_Total
              Y_Mean
              Y_Sigma
              DC_EP
              TCD_TC
              Cmin
              Cmax
              EP_LP
Test mode:    split 60% train, remainder test
```

```
=== Classifier model (full training set) ===

SMO

Classifier for classes: Y, N

BinarySMO

Machine linear: showing attribute weights, not support vectors.

        -0.1488 * (normalized) File_Name=Basal_img0019a_Combi_T1.png
 +      -1      * (normalized) File_Name=Basalioma_01_Combi_T1.png
 +      -0.0996 * (normalized) File_Name=Basalioma_02_Combi_T1.png
 +      -1      * (normalized) File_Name=Basalioma_03_Combi_T4.png
 +      -0.7022 * (normalized) File_Name=basalioma_a09f8_Combi_T1.png
 +      -0.0038 * (normalized) File_Name=Carcinoma_basal_02_Combi_T8.png
 +      -0.2634 * (normalized) File_Name=1287melanoma2_Combi_T9.png
 +      -0.2812 * (normalized) File_Name=image_a_Combi_T1.png
 +      -0.0308 * (normalized) File_Name=malig2_Combi_T1.png
 +      -0.243  * (normalized) File_Name=malignant_melanoma_2_Combi_T6.png
 +      -0.4506 * (normalized) File_Name=Melanoma_04_Combi_T6.png
 +      -0.4009 * (normalized) File_Name=Melanoma_005a_Combi_T3.png
 +      -0.6554 * (normalized) File_Name=Melanoma_006_Combi_T1.png
 +      -0.7613 * (normalized) File_Name=melanoma_007_Combi_T10.png
 +      -1      * (normalized) File_Name=melanoma_01_Combi_T6.png
 +      -0.5596 * (normalized) File_Name=melanoma_012_Combi_T1.png
 +      -0.4714 * (normalized) File_Name=melanoma_014_Combi_T1.png
 +      -0.0176 * (normalized) File_Name=Melanoma_016_Combi_T1.png
 +      -0.0856 * (normalized) File_Name=melanoma_018_Combi_T1.png
 +      -1      * (normalized) File_Name=Melanoma_02_Combi_T10.png
 +      -0.1285 * (normalized) File_Name=melanoma_a09f2_Combi_T1.png
 +      -0.2052 * (normalized) File_Name=melanoma_abdc_01_Combi_T12.png
 +      -0.1663 * (normalized) File_Name=melanoma_abdc_02_Combi_T1.png
 +      -0.3688 * (normalized) File_Name=melanoma_abdc_03_Combi_T2.png
 +      -0.3157 * (normalized) File_Name=Melanoma_img0002_Combi_T6.png
 +      -0.3478 * (normalized) File_Name=Melanoma_img0004_Combi_T1.png
```

```
 +      -0.2499 * (normalized) File_Name=Melanoma_img0005a_Combi_T1.png
 +      -0.5686 * (normalized) File_Name=Melanoma_img0033_Combi_T1.png
 +      -1      * (normalized) File_Name=Melanoma_img0048_Combi_T1.png
 +      -0.109  * (normalized) File_Name=Melanoma_img0056_Combi_T1.png
 +      -0.844  * (normalized) File_Name=Melanoma_img0081_Combi_T1.png
 +      -0.0619 * (normalized) File_Name=Melanoma_img0085a_Combi_T1.png
 +      -0.0239 * (normalized) File_Name=Melanoma_img0090_Combi_T5.png
 +      -0.2876 * (normalized) File_Name=Melanoma_img0090a_Combi_T1.png
 +      -0.1329 * (normalized) File_Name=Melanoma_img0092_Combi_T1.png
 +      -0.3605 * (normalized) File_Name=Melanoma_img0095a_Combi_T5.png
 +      -0.0661 * (normalized) File_Name=Melanoma_img0097_Combi_T1.png
 +      -0.0703 * (normalized) File_Name=Melanoma_img0105_Combi_T1.png
 +      -1      * (normalized) File_Name=melanoma_mal_Combi_T4.png
 +      -0.7867 * (normalized) File_Name=melanoma_nodule_Combi_T1.png
 +      -0.7253 * (normalized) File_Name=melanoma_palpabile_Combi_T1.png
 +      -0.2139 * (normalized) File_Name=melanoma-1_Combi_T4.png
 +      -0.7923 * (normalized) File_Name=melanoma-2_Combi_T10.png
 +      -0.1614 * (normalized) File_Name=melanoma4_Combi_T6.png
 +      -1      * (normalized) File_Name=melanoma8_Combi_T10.png
 +      -0.1177 * (normalized) File_Name=melanoma-fig3_Combi_T1.pn
 +      -0.7512 * (normalized) File_Name=melanoma-fig4_Combi_T3.pn
 +      -0.1648 * (normalized) File_Name=no_5thn_Combi_T1.png
 +      -1      * (normalized) File_Name=Escamosas_img0046_Combi_T1.png
 +       0.1381 * (normalized) File_Name=Atypical_mole_001_Combi_T6.png
 +       0.6393 * (normalized) File_Name=Atypical_mole_002_Combi_T4.png
 +       0.9589 * (normalized) File_Name=keratosis1_Combi_T1.png
 +       0.66   * (normalized) File_Name=Queratose_img0088_Combi_T4.png
 +       1      * (normalized) File_Name=Queratose_img0093_Combi_T1.png
 +       1      * (normalized) File_Name=Queratose_img0094_Combi_T1.png
 +       1      * (normalized) File_Name=Queratose_img0095_Combi_T12.png
 +                              1                          *     (normalized)
File_Name=seborrheic_keratosis_02_Combi_T1.png
 +                              1                          *     (normalized)
File_Name=seborrheic_keratosis_03_Combi_T4.png
 +       0.5909 * (normalized) File_Name=Atypical nevus_01_Combi_T1.png
 +       1      * (normalized) File_Name=Benign nevus_01_Combi_T4.png
```

```
+       0.2781 * (normalized) File_Name=Benign nevus_02_Combi_T1.png
+       1      * (normalized) File_Name=Benign nevus_03_Combi_T11.png
+       0.7963 * (normalized) File_Name=nevi4_Combi_T1.png
+       0.6617 * (normalized) File_Name=nevi4a_Combi_T1.png
+       0.5308 * (normalized) File_Name=nevo_03_Combi_T3.png
+       0.2782 * (normalized) File_Name=nevo_04_Combi_T1.png
+       0.0023 * (normalized) File_Name=nevo_congenito_Combi_T1.png
+       0.3794 * (normalized) File_Name=nevo_img0030_Combi_T1.png
+       1      * (normalized) File_Name=Nevo_img0031_Combi_T10.png
+       1      * (normalized) File_Name=nevo_img0084_Combi_T2.png
+       0.6253 * (normalized) File_Name=nevo_lentigginoso_Combi_T1.png
+       0.722  * (normalized) File_Name=nevoa3_small_Combi_T1.png
+       0.2723 * (normalized) File_Name=nevoc2_small_Combi_T1.png
+       0.1918 * (normalized) File_Name=nevodis_small_Combi_T0.png
+       0.3391 * (normalized) File_Name=nevosp1_small_Combi_T1.png
+       0.7253 * (normalized) File_Name=nevosu_small1_Combi_T3.png
+       0.1489 * (normalized) File_Name=nevou5_small_Combi_T1.png
+       0.2467 * (normalized) File_Name=Nevus_003_Combi_T1.png
+       0.0787 * (normalized) File_Name=Nevus_img0085_Combi_T4.png
+       0.0978 * (normalized) File_Name=nvmelcomp_a_Combi_T1.png
+       0.0051 * (normalized) File_Name=nvmelintra0_Combi_T1.png
+       0.7159 * (normalized) File_Name=nvmelpeq_Combi_T1.png
+       1      * (normalized) File_Name=Naevi_melanocytic3a_Combi_T1.png
+       0.6561 * (normalized) File_Name=Naevi_melanocytic3b_Combi_T1.png
+       0.0789 * (normalized) File_Name=Carolina_01
+       0.0993 * (normalized) File_Name=JCV_6mm
+       0.2781 * (normalized) File_Name=Maria_01
+       0.2534 * (normalized) Lesion_Nr_Points
+      -0.6929 * (normalized) Edge_Nr_Points
+      -0.7738 * (normalized) Dir_Changes
+      -0.7596 * (normalized) Rays_Sigma
+      -0.0789 * (normalized) C_Total
+       0.062  * (normalized) C_Mean
+      -0.5696 * (normalized) C_Sigma
+      -0.142  * (normalized) CD_Total
+      -0.2662 * (normalized) CD_Mean
```

```
 +      -0.4975 * (normalized) CD_Sigma
 +      -0.0114 * (normalized) X_Total
 +      -0.5448 * (normalized) X_Mean
 +      -0.8227 * (normalized) X_Sigma
 +      -0.0573 * (normalized) Y_Total
 +      -0.5068 * (normalized) Y_Mean
 +      -0.0144 * (normalized) Y_Sigma
 +      -0.6413 * (normalized) DC_EP
 +      -0.114  * (normalized) TCD_TC
 +       0.3124 * (normalized) Cmin
 +      -0.1937 * (normalized) Cmax
 +       0.0743 * (normalized) EP_LP
 +       1.5067

Number of kernel evaluations: 5402 (86.918% cached)



Time taken to build model: 0.07 seconds
```

```
=== Predictions on test split ===

 inst#,     actual, predicted, error, probability distribution
     1        1:Y        1:Y          *1      0
     2        2:N        1:Y      +   *1      0
     3        1:Y        1:Y          *1      0
     4        1:Y        1:Y          *1      0
     5        1:Y        2:N      +    0     *1
     6        2:N        1:Y      +   *1      0
     7        2:N        1:Y      +   *1      0
     8        1:Y        1:Y          *1      0
     9        1:Y        1:Y          *1      0
    10        2:N        2:N           0     *1
    11        2:N        2:N           0     *1
    12        1:Y        1:Y          *1      0
    13        1:Y        1:Y          *1      0
    14        2:N        2:N           0     *1
    15        1:Y        1:Y          *1      0
    16        1:Y        1:Y          *1      0
    17        2:N        2:N           0     *1
    18        1:Y        1:Y          *1      0
    19        1:Y        1:Y          *1      0
    20        1:Y        1:Y          *1      0
    21        1:Y        1:Y          *1      0
    22        1:Y        1:Y          *1      0
    23        1:Y        1:Y          *1      0
    24        2:N        2:N           0     *1
    25        2:N        1:Y      +   *1      0
    26        1:Y        1:Y          *1      0
    27        2:N        2:N           0     *1
    28        1:Y        1:Y          *1      0
    29        1:Y        1:Y          *1      0
    30        1:Y        2:N      +    0     *1
    31        2:N        2:N           0     *1
    32        1:Y        1:Y          *1      0
    33        1:Y        1:Y          *1      0
```

34	1:Y	1:Y		*1	0
35	2:N	1:Y	+	*1	0
36	1:Y	1:Y		*1	0
37	1:Y	1:Y		*1	0
38	2:N	2:N		0	*1
39	1:Y	1:Y		*1	0
40	2:N	1:Y	+	*1	0
41	1:Y	1:Y		*1	0
42	2:N	1:Y	+	*1	0
43	1:Y	1:Y		*1	0
44	2:N	1:Y	+	*1	0
45	1:Y	1:Y		*1	0
46	1:Y	1:Y		*1	0
47	2:N	2:N		0	*1
48	1:Y	1:Y		*1	0
49	2:N	2:N		0	*1
50	2:N	1:Y	+	*1	0
51	1:Y	1:Y		*1	0
52	1:Y	1:Y		*1	0
53	1:Y	1:Y		*1	0
54	1:Y	2:N	+	0	*1
55	1:Y	1:Y		*1	0

```
=== Evaluation on test split ===
=== Summary ===

Correctly Classified Instances          43               78.1818 %
Incorrectly Classified Instances        12               21.8182 %
Kappa statistic                          0.4787
K&B Relative Info Score               2757.7188 %
K&B Information Score                   24.7338 bits        0.4497
bits/instance
Class complexity | order 0              51.3348 bits        0.9334
bits/instance
Class complexity | scheme                12888         bits      234.3273
bits/instance
Complexity improvement         (Sf)         -12836.6652 bits     -233.3939
bits/instance
Mean absolute error                      0.2182
Root mean squared error                  0.4671
Relative absolute error                 49.3314 %
Root relative squared error             98.0055 %
Total Number of Instances               55

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   Class
  0.917     0.474      0.786     0.917     0.846    Y
  0.526     0.083      0.769     0.526     0.625    N

=== Confusion Matrix ===

  a  b   <-- classified as
 33  3 |  a = Y
  9 10 |  b = N
```

```
Scheme:          weka.classifiers.functions.SMO -C 1.0 -E 1.0 -G 0.01 -A
250007 -L 0.0010 -P 1.0E-12 -N 0 -V -1 -W 1
Relation:     Image_Features
Instances:    136
Attributes:   23
              File_Name
              Malignant
              Lesion_Nr_Points
              Edge_Nr_Points
              Dir_Changes
              Rays_Sigma
              C_Total
              C_Mean
              C_Sigma
              CD_Total
              CD_Mean
              CD_Sigma
              X_Total
              X_Mean
              X_Sigma
              Y_Total
              Y_Mean
              Y_Sigma
              DC_EP
              TCD_TC
              Cmin
              Cmax
              EP_LP
Test mode:    split 70% train, remainder test
```

```
=== Classifier model (full training set) ===

SMO

Classifier for classes: Y, N

BinarySMO

Machine linear: showing attribute weights, not support vectors.

        -0.1488 * (normalized) File_Name=Basal_img0019a_Combi_T1.png
 +      -1      * (normalized) File_Name=Basalioma_01_Combi_T1.png
 +      -0.0996 * (normalized) File_Name=Basalioma_02_Combi_T1.png
 +      -1      * (normalized) File_Name=Basalioma_03_Combi_T4.png
 +      -0.7022 * (normalized) File_Name=basalioma_a09f8_Combi_T1.png
 +      -0.0038 * (normalized) File_Name=Carcinoma_basal_02_Combi_T8.png
 +      -0.2634 * (normalized) File_Name=1287melanoma2_Combi_T9.png
 +      -0.2812 * (normalized) File_Name=image_a_Combi_T1.png
 +      -0.0308 * (normalized) File_Name=malig2_Combi_T1.png
 +      -0.243  * (normalized) File_Name=malignant_melanoma_2_Combi_T6.png
 +      -0.4506 * (normalized) File_Name=Melanoma_04_Combi_T6.png
 +      -0.4009 * (normalized) File_Name=Melanoma_005a_Combi_T3.png
 +      -0.6554 * (normalized) File_Name=Melanoma_006_Combi_T1.png
 +      -0.7613 * (normalized) File_Name=melanoma_007_Combi_T10.png
 +      -1      * (normalized) File_Name=melanoma_01_Combi_T6.png
 +      -0.5596 * (normalized) File_Name=melanoma_012_Combi_T1.png
 +      -0.4714 * (normalized) File_Name=melanoma_014_Combi_T1.png
 +      -0.0176 * (normalized) File_Name=Melanoma_016_Combi_T1.png
 +      -0.0856 * (normalized) File_Name=melanoma_018_Combi_T1.png
 +      -1      * (normalized) File_Name=Melanoma_02_Combi_T10.png
 +      -0.1285 * (normalized) File_Name=melanoma_a09f2_Combi_T1.png
 +      -0.2052 * (normalized) File_Name=melanoma_abdc_01_Combi_T12.png
 +      -0.1663 * (normalized) File_Name=melanoma_abdc_02_Combi_T1.png
 +      -0.3688 * (normalized) File_Name=melanoma_abdc_03_Combi_T2.png
 +      -0.3157 * (normalized) File_Name=Melanoma_img0002_Combi_T6.png
 +      -0.3478 * (normalized) File_Name=Melanoma_img0004_Combi_T1.png
```

```
 +      -0.2499 * (normalized) File_Name=Melanoma_img0005a_Combi_T1.png
 +      -0.5686 * (normalized) File_Name=Melanoma_img0033_Combi_T1.png
 +      -1      * (normalized) File_Name=Melanoma_img0048_Combi_T1.png
 +      -0.109  * (normalized) File_Name=Melanoma_img0056_Combi_T1.png
 +      -0.844  * (normalized) File_Name=Melanoma_img0081_Combi_T1.png
 +      -0.0619 * (normalized) File_Name=Melanoma_img0085a_Combi_T1.png
 +      -0.0239 * (normalized) File_Name=Melanoma_img0090_Combi_T5.png
 +      -0.2876 * (normalized) File_Name=Melanoma_img0090a_Combi_T1.png
 +      -0.1329 * (normalized) File_Name=Melanoma_img0092_Combi_T1.png
 +      -0.3605 * (normalized) File_Name=Melanoma_img0095a_Combi_T5.png
 +      -0.0661 * (normalized) File_Name=Melanoma_img0097_Combi_T1.png
 +      -0.0703 * (normalized) File_Name=Melanoma_img0105_Combi_T1.png
 +      -1      * (normalized) File_Name=melanoma_mal_Combi_T4.png
 +      -0.7867 * (normalized) File_Name=melanoma_nodule_Combi_T1.png
 +      -0.7253 * (normalized) File_Name=melanoma_palpabile_Combi_T1.png
 +      -0.2139 * (normalized) File_Name=melanoma-1_Combi_T4.png
 +      -0.7923 * (normalized) File_Name=melanoma-2_Combi_T10.png
 +      -0.1614 * (normalized) File_Name=melanoma4_Combi_T6.png
 +      -1      * (normalized) File_Name=melanoma8_Combi_T10.png
 +      -0.1177 * (normalized) File_Name=melanoma-fig3_Combi_T1.pn
 +      -0.7512 * (normalized) File_Name=melanoma-fig4_Combi_T3.pn
 +      -0.1648 * (normalized) File_Name=no_5thn_Combi_T1.png
 +      -1      * (normalized) File_Name=Escamosas_img0046_Combi_T1.png
 +       0.1381 * (normalized) File_Name=Atypical_mole_001_Combi_T6.png
 +       0.6393 * (normalized) File_Name=Atypical_mole_002_Combi_T4.png
 +       0.9589 * (normalized) File_Name=keratosis1_Combi_T1.png
 +       0.66   * (normalized) File_Name=Queratose_img0088_Combi_T4.png
 +       1      * (normalized) File_Name=Queratose_img0093_Combi_T1.png
 +       1      * (normalized) File_Name=Queratose_img0094_Combi_T1.png
 +       1      * (normalized) File_Name=Queratose_img0095_Combi_T12.png
 +                              1                          *     (normalized)
File_Name=seborrheic_keratosis_02_Combi_T1.png
 +                              1                          *     (normalized)
File_Name=seborrheic_keratosis_03_Combi_T4.png
 +       0.5909 * (normalized) File_Name=Atypical nevus_01_Combi_T1.png
 +       1      * (normalized) File_Name=Benign nevus_01_Combi_T4.png
```

```
+       0.2781 * (normalized) File_Name=Benign nevus_02_Combi_T1.png
+       1      * (normalized) File_Name=Benign nevus_03_Combi_T11.png
+       0.7963 * (normalized) File_Name=nevi4_Combi_T1.png
+       0.6617 * (normalized) File_Name=nevi4a_Combi_T1.png
+       0.5308 * (normalized) File_Name=nevo_03_Combi_T3.png
+       0.2782 * (normalized) File_Name=nevo_04_Combi_T1.png
+       0.0023 * (normalized) File_Name=nevo_congenito_Combi_T1.png
+       0.3794 * (normalized) File_Name=nevo_img0030_Combi_T1.png
+       1      * (normalized) File_Name=Nevo_img0031_Combi_T10.png
+       1      * (normalized) File_Name=nevo_img0084_Combi_T2.png
+       0.6253 * (normalized) File_Name=nevo_lentigginoso_Combi_T1.png
+       0.722  * (normalized) File_Name=nevoa3_small_Combi_T1.png
+       0.2723 * (normalized) File_Name=nevoc2_small_Combi_T1.png
+       0.1918 * (normalized) File_Name=nevodis_small_Combi_T0.png
+       0.3391 * (normalized) File_Name=nevosp1_small_Combi_T1.png
+       0.7253 * (normalized) File_Name=nevosu_small1_Combi_T3.png
+       0.1489 * (normalized) File_Name=nevou5_small_Combi_T1.png
+       0.2467 * (normalized) File_Name=Nevus_003_Combi_T1.png
+       0.0787 * (normalized) File_Name=Nevus_img0085_Combi_T4.png
+       0.0978 * (normalized) File_Name=nvmelcomp_a_Combi_T1.png
+       0.0051 * (normalized) File_Name=nvmelintra0_Combi_T1.png
+       0.7159 * (normalized) File_Name=nvmelpeq_Combi_T1.png
+       1      * (normalized) File_Name=Naevi_melanocytic3a_Combi_T1.png
+       0.6561 * (normalized) File_Name=Naevi_melanocytic3b_Combi_T1.png
+       0.0789 * (normalized) File_Name=Carolina_01
+       0.0993 * (normalized) File_Name=JCV_6mm
+       0.2781 * (normalized) File_Name=Maria_01
+       0.2534 * (normalized) Lesion_Nr_Points
+      -0.6929 * (normalized) Edge_Nr_Points
+      -0.7738 * (normalized) Dir_Changes
+      -0.7596 * (normalized) Rays_Sigma
+      -0.0789 * (normalized) C_Total
+       0.062  * (normalized) C_Mean
+      -0.5696 * (normalized) C_Sigma
+      -0.142  * (normalized) CD_Total
+      -0.2662 * (normalized) CD_Mean
```

```
 +      -0.4975 * (normalized) CD_Sigma
 +      -0.0114 * (normalized) X_Total
 +      -0.5448 * (normalized) X_Mean
 +      -0.8227 * (normalized) X_Sigma
 +      -0.0573 * (normalized) Y_Total
 +      -0.5068 * (normalized) Y_Mean
 +      -0.0144 * (normalized) Y_Sigma
 +      -0.6413 * (normalized) DC_EP
 +      -0.114  * (normalized) TCD_TC
 +       0.3124 * (normalized) Cmin
 +      -0.1937 * (normalized) Cmax
 +       0.0743 * (normalized) EP_LP
 +       1.5067

Number of kernel evaluations: 5402 (86.918% cached)



Time taken to build model: 0.07 seconds
```

```
=== Predictions on test split ===

 inst#,     actual, predicted, error, probability distribution
     1        1:Y        1:Y          *1      0
     2        1:Y        1:Y          *1      0
     3        2:N        2:N           0     *1
     4        1:Y        1:Y          *1      0
     5        1:Y        1:Y          *1      0
     6        1:Y        1:Y          *1      0
     7        1:Y        1:Y          *1      0
     8        1:Y        1:Y          *1      0
     9        1:Y        1:Y          *1      0
    10        2:N        2:N           0     *1
    11        2:N        1:Y      +   *1      0
    12        1:Y        1:Y          *1      0
    13        2:N        2:N           0     *1
    14        1:Y        1:Y          *1      0
    15        1:Y        1:Y          *1      0
    16        1:Y        2:N      +    0     *1
    17        2:N        2:N           0     *1
    18        1:Y        1:Y          *1      0
    19        1:Y        1:Y          *1      0
    20        1:Y        1:Y          *1      0
    21        2:N        1:Y      +   *1      0
    22        1:Y        1:Y          *1      0
    23        1:Y        1:Y          *1      0
    24        2:N        2:N           0     *1
    25        1:Y        1:Y          *1      0
    26        2:N        1:Y      +   *1      0
    27        1:Y        1:Y          *1      0
    28        2:N        1:Y      +   *1      0
    29        1:Y        1:Y          *1      0
    30        2:N        1:Y      +   *1      0
    31        1:Y        1:Y          *1      0
    32        1:Y        1:Y          *1      0
    33        2:N        2:N           0     *1
```

34	1:Y	1:Y		*1	0
35	2:N	2:N		0	*1
36	2:N	2:N		0	*1
37	1:Y	1:Y		*1	0
38	1:Y	1:Y		*1	0
39	1:Y	1:Y		*1	0
40	1:Y	2:N	+	0	*1
41	1:Y	1:Y		*1	0

```
=== Evaluation on test split ===
=== Summary ===

Correctly Classified Instances          34               82.9268 %
Incorrectly Classified Instances         7               17.0732 %
Kappa statistic                          0.5798
K&B Relative Info Score               2375.101  %
K&B Information Score                           21.7279 bits        0.5299
bits/instance
Class complexity | order 0                      36.9701 bits        0.9017
bits/instance
Class complexity | scheme                     7518         bits     183.3659
bits/instance
Complexity improvement         (Sf)           -7481.0299 bits     -182.4641
bits/instance
Mean absolute error                      0.1707
Root mean squared error                  0.4132
Relative absolute error                 39.0006 %
Root relative squared error             88.7617 %
Total Number of Instances               41

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   Class
  0.929     0.385      0.839     0.929     0.881    Y
  0.615     0.071      0.8       0.615     0.696    N

=== Confusion Matrix ===

  a  b   <-- classified as
 26  2 |  a = Y
  5  8 |  b = N
```

```
Scheme:          weka.classifiers.functions.SMO -C 1.0 -E 1.0 -G 0.01 -A
250007 -L 0.0010 -P 1.0E-12 -N 0 -V -1 -W 1
Relation:     Image_Features
Instances:    136
Attributes:   23
              File_Name
              Malignant
              Lesion_Nr_Points
              Edge_Nr_Points
              Dir_Changes
              Rays_Sigma
              C_Total
              C_Mean
              C_Sigma
              CD_Total
              CD_Mean
              CD_Sigma
              X_Total
              X_Mean
              X_Sigma
              Y_Total
              Y_Mean
              Y_Sigma
              DC_EP
              TCD_TC
              Cmin
              Cmax
              EP_LP
Test mode:    split 80% train, remainder test

=== Classifier model (full training set) ===

SMO

Classifier for classes: Y, N
```

```
BinarySMO

Machine linear: showing attribute weights, not support vectors.

        -0.1488 * (normalized) File_Name=Basal_img0019a_Combi_T1.png
 +      -1      * (normalized) File_Name=Basalioma_01_Combi_T1.png
 +      -0.0996 * (normalized) File_Name=Basalioma_02_Combi_T1.png
 +      -1      * (normalized) File_Name=Basalioma_03_Combi_T4.png
 +      -0.7022 * (normalized) File_Name=basalioma_a09f8_Combi_T1.png
 +      -0.0038 * (normalized) File_Name=Carcinoma_basal_02_Combi_T8.png
 +      -0.2634 * (normalized) File_Name=1287melanoma2_Combi_T9.png
 +      -0.2812 * (normalized) File_Name=image_a_Combi_T1.png
 +      -0.0308 * (normalized) File_Name=malig2_Combi_T1.png
 +      -0.243  * (normalized) File_Name=malignant_melanoma_2_Combi_T6.png
 +      -0.4506 * (normalized) File_Name=Melanoma_04_Combi_T6.png
 +      -0.4009 * (normalized) File_Name=Melanoma_005a_Combi_T3.png
 +      -0.6554 * (normalized) File_Name=Melanoma_006_Combi_T1.png
 +      -0.7613 * (normalized) File_Name=melanoma_007_Combi_T10.png
 +      -1      * (normalized) File_Name=melanoma_01_Combi_T6.png
 +      -0.5596 * (normalized) File_Name=melanoma_012_Combi_T1.png
 +      -0.4714 * (normalized) File_Name=melanoma_014_Combi_T1.png
 +      -0.0176 * (normalized) File_Name=Melanoma_016_Combi_T1.png
 +      -0.0856 * (normalized) File_Name=melanoma_018_Combi_T1.png
 +      -1      * (normalized) File_Name=Melanoma_02_Combi_T10.png
 +      -0.1285 * (normalized) File_Name=melanoma_a09f2_Combi_T1.png
 +      -0.2052 * (normalized) File_Name=melanoma_abdc_01_Combi_T12.png
 +      -0.1663 * (normalized) File_Name=melanoma_abdc_02_Combi_T1.png
 +      -0.3688 * (normalized) File_Name=melanoma_abdc_03_Combi_T2.png
 +      -0.3157 * (normalized) File_Name=Melanoma_img0002_Combi_T6.png
 +      -0.3478 * (normalized) File_Name=Melanoma_img0004_Combi_T1.png
 +      -0.2499 * (normalized) File_Name=Melanoma_img0005a_Combi_T1.png
 +      -0.5686 * (normalized) File_Name=Melanoma_img0033_Combi_T1.png
 +      -1      * (normalized) File_Name=Melanoma_img0048_Combi_T1.png
 +      -0.109  * (normalized) File_Name=Melanoma_img0056_Combi_T1.png
 +      -0.844  * (normalized) File_Name=Melanoma_img0081_Combi_T1.png
 +      -0.0619 * (normalized) File_Name=Melanoma_img0085a_Combi_T1.png
```

```
 +      -0.0239 * (normalized) File_Name=Melanoma_img0090_Combi_T5.png
 +      -0.2876 * (normalized) File_Name=Melanoma_img0090a_Combi_T1.png
 +      -0.1329 * (normalized) File_Name=Melanoma_img0092_Combi_T1.png
 +      -0.3605 * (normalized) File_Name=Melanoma_img0095a_Combi_T5.png
 +      -0.0661 * (normalized) File_Name=Melanoma_img0097_Combi_T1.png
 +      -0.0703 * (normalized) File_Name=Melanoma_img0105_Combi_T1.png
 +      -1      * (normalized) File_Name=melanoma_mal_Combi_T4.png
 +      -0.7867 * (normalized) File_Name=melanoma_nodule_Combi_T1.png
 +      -0.7253 * (normalized) File_Name=melanoma_palpabile_Combi_T1.png
 +      -0.2139 * (normalized) File_Name=melanoma-1_Combi_T4.png
 +      -0.7923 * (normalized) File_Name=melanoma-2_Combi_T10.png
 +      -0.1614 * (normalized) File_Name=melanoma4_Combi_T6.png
 +      -1      * (normalized) File_Name=melanoma8_Combi_T10.png
 +      -0.1177 * (normalized) File_Name=melanoma-fig3_Combi_T1.pn
 +      -0.7512 * (normalized) File_Name=melanoma-fig4_Combi_T3.pn
 +      -0.1648 * (normalized) File_Name=no_5thn_Combi_T1.png
 +      -1      * (normalized) File_Name=Escamosas_img0046_Combi_T1.png
 +       0.1381 * (normalized) File_Name=Atypical_mole_001_Combi_T6.png
 +       0.6393 * (normalized) File_Name=Atypical_mole_002_Combi_T4.png
 +       0.9589 * (normalized) File_Name=keratosis1_Combi_T1.png
 +       0.66   * (normalized) File_Name=Queratose_img0088_Combi_T4.png
 +       1      * (normalized) File_Name=Queratose_img0093_Combi_T1.png
 +       1      * (normalized) File_Name=Queratose_img0094_Combi_T1.png
 +       1      * (normalized) File_Name=Queratose_img0095_Combi_T12.png
 +                              1                          *     (normalized)
File_Name=seborrheic_keratosis_02_Combi_T1.png
 +                              1                          *     (normalized)
File_Name=seborrheic_keratosis_03_Combi_T4.png
 +       0.5909 * (normalized) File_Name=Atypical nevus_01_Combi_T1.png
 +       1      * (normalized) File_Name=Benign nevus_01_Combi_T4.png
 +       0.2781 * (normalized) File_Name=Benign nevus_02_Combi_T1.png
 +       1      * (normalized) File_Name=Benign nevus_03_Combi_T11.png
 +       0.7963 * (normalized) File_Name=nevi4_Combi_T1.png
 +       0.6617 * (normalized) File_Name=nevi4a_Combi_T1.png
 +       0.5308 * (normalized) File_Name=nevo_03_Combi_T3.png
 +       0.2782 * (normalized) File_Name=nevo_04_Combi_T1.png
```

```
+       0.0023 * (normalized) File_Name=nevo_congenito_Combi_T1.png
+       0.3794 * (normalized) File_Name=nevo_img0030_Combi_T1.png
+       1      * (normalized) File_Name=Nevo_img0031_Combi_T10.png
+       1      * (normalized) File_Name=nevo_img0084_Combi_T2.png
+       0.6253 * (normalized) File_Name=nevo_lentigginoso_Combi_T1.png
+       0.722  * (normalized) File_Name=nevoa3_small_Combi_T1.png
+       0.2723 * (normalized) File_Name=nevoc2_small_Combi_T1.png
+       0.1918 * (normalized) File_Name=nevodis_small_Combi_T0.png
+       0.3391 * (normalized) File_Name=nevosp1_small_Combi_T1.png
+       0.7253 * (normalized) File_Name=nevosu_small1_Combi_T3.png
+       0.1489 * (normalized) File_Name=nevou5_small_Combi_T1.png
+       0.2467 * (normalized) File_Name=Nevus_003_Combi_T1.png
+       0.0787 * (normalized) File_Name=Nevus_img0085_Combi_T4.png
+       0.0978 * (normalized) File_Name=nvmelcomp_a_Combi_T1.png
+       0.0051 * (normalized) File_Name=nvmelintra0_Combi_T1.png
+       0.7159 * (normalized) File_Name=nvmelpeq_Combi_T1.png
+       1      * (normalized) File_Name=Naevi_melanocytic3a_Combi_T1.png
+       0.6561 * (normalized) File_Name=Naevi_melanocytic3b_Combi_T1.png
+       0.0789 * (normalized) File_Name=Carolina_01
+       0.0993 * (normalized) File_Name=JCV_6mm
+       0.2781 * (normalized) File_Name=Maria_01
+       0.2534 * (normalized) Lesion_Nr_Points
+      -0.6929 * (normalized) Edge_Nr_Points
+      -0.7738 * (normalized) Dir_Changes
+      -0.7596 * (normalized) Rays_Sigma
+      -0.0789 * (normalized) C_Total
+       0.062  * (normalized) C_Mean
+      -0.5696 * (normalized) C_Sigma
+      -0.142  * (normalized) CD_Total
+      -0.2662 * (normalized) CD_Mean
+      -0.4975 * (normalized) CD_Sigma
+      -0.0114 * (normalized) X_Total
+      -0.5448 * (normalized) X_Mean
+      -0.8227 * (normalized) X_Sigma
+      -0.0573 * (normalized) Y_Total
+      -0.5068 * (normalized) Y_Mean
```

```
 +      -0.0144 * (normalized) Y_Sigma
 +      -0.6413 * (normalized) DC_EP
 +      -0.114  * (normalized) TCD_TC
 +       0.3124 * (normalized) Cmin
 +      -0.1937 * (normalized) Cmax
 +       0.0743 * (normalized) EP_LP
 +       1.5067

Number of kernel evaluations: 5402 (86.918% cached)



Time taken to build model: 0.13 seconds
```

```
=== Predictions on test split ===

 inst#,     actual, predicted, error, probability distribution
     1        1:Y        1:Y          *1      0
     2        1:Y        1:Y          *1      0
     3        1:Y        2:N      +    0     *1
     4        2:N        2:N           0     *1
     5        1:Y        1:Y          *1      0
     6        1:Y        1:Y          *1      0
     7        1:Y        1:Y          *1      0
     8        2:N        1:Y      +   *1      0
     9        1:Y        1:Y          *1      0
    10        1:Y        1:Y          *1      0
    11        2:N        2:N           0     *1
    12        1:Y        1:Y          *1      0
    13        2:N        1:Y      +   *1      0
    14        1:Y        1:Y          *1      0
    15        2:N        1:Y      +   *1      0
    16        1:Y        1:Y          *1      0
    17        2:N        1:Y      +   *1      0
    18        1:Y        1:Y          *1      0
    19        1:Y        1:Y          *1      0
    20        2:N        2:N           0     *1
    21        1:Y        1:Y          *1      0
    22        2:N        2:N           0     *1
    23        2:N        2:N           0     *1
    24        1:Y        1:Y          *1      0
    25        1:Y        1:Y          *1      0
    26        1:Y        1:Y          *1      0
    27        1:Y        2:N      +    0     *1
    28        1:Y        1:Y          *1      0
```

```
=== Evaluation on test split ===
=== Summary ===

Correctly Classified Instances          22               78.5714 %
Incorrectly Classified Instances         6               21.4286 %
Kappa statistic                          0.4783
K&B Relative Info Score               1345.124  %
K&B Information Score                           12.2691  bits         0.4382
bits/instance
Class complexity | order 0                      25.3691  bits          0.906
bits/instance
Class complexity | scheme                     6444         bits      230.1429
bits/instance
Complexity improvement         (Sf)           -6418.6309  bits     -229.2368
bits/instance
Mean absolute error                      0.2143
Root mean squared error                  0.4629
Relative absolute error                 48.8889 %
Root relative squared error             99.1112 %
Total Number of Instances               28

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   Class
  0.895     0.444      0.81      0.895     0.85     Y
  0.556     0.105      0.714     0.556     0.625    N


=== Confusion Matrix ===

  a  b   <-- classified as
 17  2 |  a = Y
  4  5 |  b = N
```

```
Scheme:          weka.classifiers.functions.SMO -C 1.0 -E 1.0 -G 0.01 -A
250007 -L 0.0010 -P 1.0E-12 -N 0 -V -1 -W 1
Relation:     Image_Features
Instances:    136
Attributes:   23
              File_Name
              Malignant
              Lesion_Nr_Points
              Edge_Nr_Points
              Dir_Changes
              Rays_Sigma
              C_Total
              C_Mean
              C_Sigma
              CD_Total
              CD_Mean
              CD_Sigma
              X_Total
              X_Mean
              X_Sigma
              Y_Total
              Y_Mean
              Y_Sigma
              DC_EP
              TCD_TC
              Cmin
              Cmax
              EP_LP
Test mode:    split 80% train, remainder test

=== Classifier model (full training set) ===

SMO

Classifier for classes: Y, N
```

```
BinarySMO

Machine linear: showing attribute weights, not support vectors.

        -0.1488 * (normalized) File_Name=Basal_img0019a_Combi_T1.png
 +      -1      * (normalized) File_Name=Basalioma_01_Combi_T1.png
 +      -0.0996 * (normalized) File_Name=Basalioma_02_Combi_T1.png
 +      -1      * (normalized) File_Name=Basalioma_03_Combi_T4.png
 +      -0.7022 * (normalized) File_Name=basalioma_a09f8_Combi_T1.png
 +      -0.0038 * (normalized) File_Name=Carcinoma_basal_02_Combi_T8.png
 +      -0.2634 * (normalized) File_Name=1287melanoma2_Combi_T9.png
 +      -0.2812 * (normalized) File_Name=image_a_Combi_T1.png
 +      -0.0308 * (normalized) File_Name=malig2_Combi_T1.png
 +      -0.243  * (normalized) File_Name=malignant_melanoma_2_Combi_T6.png
 +      -0.4506 * (normalized) File_Name=Melanoma_04_Combi_T6.png
 +      -0.4009 * (normalized) File_Name=Melanoma_005a_Combi_T3.png
 +      -0.6554 * (normalized) File_Name=Melanoma_006_Combi_T1.png
 +      -0.7613 * (normalized) File_Name=melanoma_007_Combi_T10.png
 +      -1      * (normalized) File_Name=melanoma_01_Combi_T6.png
 +      -0.5596 * (normalized) File_Name=melanoma_012_Combi_T1.png
 +      -0.4714 * (normalized) File_Name=melanoma_014_Combi_T1.png
 +      -0.0176 * (normalized) File_Name=Melanoma_016_Combi_T1.png
 +      -0.0856 * (normalized) File_Name=melanoma_018_Combi_T1.png
 +      -1      * (normalized) File_Name=Melanoma_02_Combi_T10.png
 +      -0.1285 * (normalized) File_Name=melanoma_a09f2_Combi_T1.png
 +      -0.2052 * (normalized) File_Name=melanoma_abdc_01_Combi_T12.png
 +      -0.1663 * (normalized) File_Name=melanoma_abdc_02_Combi_T1.png
 +      -0.3688 * (normalized) File_Name=melanoma_abdc_03_Combi_T2.png
 +      -0.3157 * (normalized) File_Name=Melanoma_img0002_Combi_T6.png
 +      -0.3478 * (normalized) File_Name=Melanoma_img0004_Combi_T1.png
 +      -0.2499 * (normalized) File_Name=Melanoma_img0005a_Combi_T1.png
 +      -0.5686 * (normalized) File_Name=Melanoma_img0033_Combi_T1.png
 +      -1      * (normalized) File_Name=Melanoma_img0048_Combi_T1.png
 +      -0.109  * (normalized) File_Name=Melanoma_img0056_Combi_T1.png
 +      -0.844  * (normalized) File_Name=Melanoma_img0081_Combi_T1.png
 +      -0.0619 * (normalized) File_Name=Melanoma_img0085a_Combi_T1.png
```

```
 +      -0.0239 * (normalized) File_Name=Melanoma_img0090_Combi_T5.png
 +      -0.2876 * (normalized) File_Name=Melanoma_img0090a_Combi_T1.png
 +      -0.1329 * (normalized) File_Name=Melanoma_img0092_Combi_T1.png
 +      -0.3605 * (normalized) File_Name=Melanoma_img0095a_Combi_T5.png
 +      -0.0661 * (normalized) File_Name=Melanoma_img0097_Combi_T1.png
 +      -0.0703 * (normalized) File_Name=Melanoma_img0105_Combi_T1.png
 +      -1      * (normalized) File_Name=melanoma_mal_Combi_T4.png
 +      -0.7867 * (normalized) File_Name=melanoma_nodule_Combi_T1.png
 +      -0.7253 * (normalized) File_Name=melanoma_palpabile_Combi_T1.png
 +      -0.2139 * (normalized) File_Name=melanoma-1_Combi_T4.png
 +      -0.7923 * (normalized) File_Name=melanoma-2_Combi_T10.png
 +      -0.1614 * (normalized) File_Name=melanoma4_Combi_T6.png
 +      -1      * (normalized) File_Name=melanoma8_Combi_T10.png
 +      -0.1177 * (normalized) File_Name=melanoma-fig3_Combi_T1.pn
 +      -0.7512 * (normalized) File_Name=melanoma-fig4_Combi_T3.pn
 +      -0.1648 * (normalized) File_Name=no_5thn_Combi_T1.png
 +      -1      * (normalized) File_Name=Escamosas_img0046_Combi_T1.png
 +       0.1381 * (normalized) File_Name=Atypical_mole_001_Combi_T6.png
 +       0.6393 * (normalized) File_Name=Atypical_mole_002_Combi_T4.png
 +       0.9589 * (normalized) File_Name=keratosis1_Combi_T1.png
 +       0.66   * (normalized) File_Name=Queratose_img0088_Combi_T4.png
 +       1      * (normalized) File_Name=Queratose_img0093_Combi_T1.png
 +       1      * (normalized) File_Name=Queratose_img0094_Combi_T1.png
 +       1      * (normalized) File_Name=Queratose_img0095_Combi_T12.png
 +                              1                          *     (normalized)
File_Name=seborrheic_keratosis_02_Combi_T1.png
 +                              1                          *     (normalized)
File_Name=seborrheic_keratosis_03_Combi_T4.png
 +       0.5909 * (normalized) File_Name=Atypical nevus_01_Combi_T1.png
 +       1      * (normalized) File_Name=Benign nevus_01_Combi_T4.png
 +       0.2781 * (normalized) File_Name=Benign nevus_02_Combi_T1.png
 +       1      * (normalized) File_Name=Benign nevus_03_Combi_T11.png
 +       0.7963 * (normalized) File_Name=nevi4_Combi_T1.png
 +       0.6617 * (normalized) File_Name=nevi4a_Combi_T1.png
 +       0.5308 * (normalized) File_Name=nevo_03_Combi_T3.png
 +       0.2782 * (normalized) File_Name=nevo_04_Combi_T1.png
```

```
+       0.0023 * (normalized) File_Name=nevo_congenito_Combi_T1.png
+       0.3794 * (normalized) File_Name=nevo_img0030_Combi_T1.png
+       1      * (normalized) File_Name=Nevo_img0031_Combi_T10.png
+       1      * (normalized) File_Name=nevo_img0084_Combi_T2.png
+       0.6253 * (normalized) File_Name=nevo_lentigginoso_Combi_T1.png
+       0.722  * (normalized) File_Name=nevoa3_small_Combi_T1.png
+       0.2723 * (normalized) File_Name=nevoc2_small_Combi_T1.png
+       0.1918 * (normalized) File_Name=nevodis_small_Combi_T0.png
+       0.3391 * (normalized) File_Name=nevosp1_small_Combi_T1.png
+       0.7253 * (normalized) File_Name=nevosu_small1_Combi_T3.png
+       0.1489 * (normalized) File_Name=nevou5_small_Combi_T1.png
+       0.2467 * (normalized) File_Name=Nevus_003_Combi_T1.png
+       0.0787 * (normalized) File_Name=Nevus_img0085_Combi_T4.png
+       0.0978 * (normalized) File_Name=nvmelcomp_a_Combi_T1.png
+       0.0051 * (normalized) File_Name=nvmelintra0_Combi_T1.png
+       0.7159 * (normalized) File_Name=nvmelpeq_Combi_T1.png
+       1      * (normalized) File_Name=Naevi_melanocytic3a_Combi_T1.png
+       0.6561 * (normalized) File_Name=Naevi_melanocytic3b_Combi_T1.png
+       0.0789 * (normalized) File_Name=Carolina_01
+       0.0993 * (normalized) File_Name=JCV_6mm
+       0.2781 * (normalized) File_Name=Maria_01
+       0.2534 * (normalized) Lesion_Nr_Points
+      -0.6929 * (normalized) Edge_Nr_Points
+      -0.7738 * (normalized) Dir_Changes
+      -0.7596 * (normalized) Rays_Sigma
+      -0.0789 * (normalized) C_Total
+       0.062  * (normalized) C_Mean
+      -0.5696 * (normalized) C_Sigma
+      -0.142  * (normalized) CD_Total
+      -0.2662 * (normalized) CD_Mean
+      -0.4975 * (normalized) CD_Sigma
+      -0.0114 * (normalized) X_Total
+      -0.5448 * (normalized) X_Mean
+      -0.8227 * (normalized) X_Sigma
+      -0.0573 * (normalized) Y_Total
+      -0.5068 * (normalized) Y_Mean
```

```
 +      -0.0144 * (normalized) Y_Sigma
 +      -0.6413 * (normalized) DC_EP
 +      -0.114  * (normalized) TCD_TC
 +       0.3124 * (normalized) Cmin
 +      -0.1937 * (normalized) Cmax
 +       0.0743 * (normalized) EP_LP
 +       1.5067

Number of kernel evaluations: 5402 (86.918% cached)



Time taken to build model: 0.13 seconds
```

```
=== Predictions on test split ===

 inst#,     actual, predicted, error, probability distribution
     1        1:Y        1:Y          *1      0
     2        1:Y        1:Y          *1      0
     3        1:Y        2:N      +    0     *1
     4        2:N        2:N           0     *1
     5        1:Y        1:Y          *1      0
     6        1:Y        1:Y          *1      0
     7        1:Y        1:Y          *1      0
     8        2:N        1:Y      +   *1      0
     9        1:Y        1:Y          *1      0
    10        1:Y        1:Y          *1      0
    11        2:N        2:N           0     *1
    12        1:Y        1:Y          *1      0
    13        2:N        1:Y      +   *1      0
    14        1:Y        1:Y          *1      0
    15        2:N        1:Y      +   *1      0
    16        1:Y        1:Y          *1      0
    17        2:N        1:Y      +   *1      0
    18        1:Y        1:Y          *1      0
    19        1:Y        1:Y          *1      0
    20        2:N        2:N           0     *1
    21        1:Y        1:Y          *1      0
    22        2:N        2:N           0     *1
    23        2:N        2:N           0     *1
    24        1:Y        1:Y          *1      0
    25        1:Y        1:Y          *1      0
    26        1:Y        1:Y          *1      0
    27        1:Y        2:N      +    0     *1
    28        1:Y        1:Y          *1      0
```

```
=== Evaluation on test split ===
=== Summary ===

Correctly Classified Instances          22               78.5714 %
Incorrectly Classified Instances         6               21.4286 %
Kappa statistic                          0.4783
K&B Relative Info Score               1345.124  %
K&B Information Score                           12.2691  bits         0.4382
bits/instance
Class complexity | order 0                      25.3691  bits          0.906
bits/instance
Class complexity | scheme                     6444         bits      230.1429
bits/instance
Complexity improvement         (Sf)           -6418.6309  bits     -229.2368
bits/instance
Mean absolute error                      0.2143
Root mean squared error                  0.4629
Relative absolute error                 48.8889 %
Root relative squared error             99.1112 %
Total Number of Instances               28

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   Class
  0.895     0.444      0.81      0.895     0.85     Y
  0.556     0.105      0.714     0.556     0.625    N


=== Confusion Matrix ===

  a  b   <-- classified as
 17  2 |  a = Y
  4  5 |  b = N
```

```
Scheme:       weka.classifiers.functions.MultilayerPerceptron -L 0.3 -M 0.2
-N 500 -V 0 -S 0 -E 20 -H a
Relation:     Image_Features
Instances:    136
Attributes:   23
              File_Name
              Malignant
              Lesion_Nr_Points
              Edge_Nr_Points
              Dir_Changes
              Rays_Sigma
              C_Total
              C_Mean
              C_Sigma
              CD_Total
              CD_Mean
              CD_Sigma
              X_Total
              X_Mean
              X_Sigma
              Y_Total
              Y_Mean
              Y_Sigma
              DC_EP
              TCD_TC
              Cmin
              Cmax
              EP_LP
Test mode:    split 50% train, remainder test
Class Y
    Input
    Node 0
Class N
    Input
    Node 1
Time taken to build model: 113.95 seconds
```

```
=== Predictions on test split ===

 inst#,     actual, predicted, error, probability distribution
     1        1:Y        1:Y          *0.999  0.001
     2        2:N        2:N           0.04  *0.96
     3        1:Y        1:Y          *0.799  0.201
     4        1:Y        1:Y          *0.78   0.22
     5        2:N        1:Y      +   *0.976  0.024
     6        2:N        2:N           0.002 *0.998
     7        2:N        1:Y      +   *0.935  0.065
     8        1:Y        1:Y          *0.984  0.016
     9        1:Y        2:N      +    0.021 *0.979
    10        1:Y        2:N      +    0.496 *0.504
    11        1:Y        2:N      +    0.084 *0.916
    12        1:Y        1:Y          *0.919  0.081
    13        1:Y        1:Y          *0.998  0.002
    14        1:Y        1:Y          *0.964  0.036
    15        1:Y        1:Y          *1      0
    16        2:N        2:N           0.002 *0.998
    17        1:Y        1:Y          *0.999  0.001
    18        2:N        2:N           0     *1
    19        1:Y        1:Y          *0.998  0.002
    20        1:Y        1:Y          *0.885  0.115
    21        1:Y        1:Y          *1      0
    22        2:N        2:N           0.115 *0.885
    23        1:Y        2:N      +    0.119 *0.881
    24        1:Y        1:Y          *1      0
    25        1:Y        1:Y          *0.994  0.006
    26        1:Y        1:Y          *0.999  0.001
    27        2:N        2:N           0     *1
    28        1:Y        1:Y          *0.992  0.008
    29        1:Y        2:N      +    0.141 *0.859
    30        2:N        2:N           0     *1
    31        2:N        2:N           0.001 *0.999
    32        1:Y        1:Y          *0.998  0.002
    33        2:N        1:Y      +   *0.73   0.27
```

```
34       2:N       2:N          0.009 *0.991
35       1:Y       1:Y        *1      0
36       2:N       2:N          0.032 *0.968
37       1:Y       2:N      +   0.006 *0.994
38       1:Y       1:Y        *0.846  0.154
39       1:Y       2:N      +   0.286 *0.714
40       2:N       2:N          0.001 *0.999
41       1:Y       2:N      +   0.265 *0.735
42       2:N       2:N          0.001 *0.999
43       1:Y       1:Y        *1      0
44       1:Y       1:Y        *1      0
45       2:N       2:N          0.003 *0.997
46       2:N       2:N          0.001 *0.999
47       2:N       2:N          0.016 *0.984
48       2:N       2:N          0.108 *0.892
49       1:Y       1:Y        *0.999  0.001
50       1:Y       1:Y        *0.945  0.055
51       1:Y       1:Y        *0.999  0.001
52       2:N       2:N          0.018 *0.982
53       2:N       2:N          0.288 *0.712
54       1:Y       1:Y        *1      0
55       1:Y       1:Y        *1      0
56       1:Y       1:Y        *1      0
57       2:N       2:N          0.371 *0.629
58       2:N       1:Y      +  *0.784  0.216
59       1:Y       1:Y        *1      0
60       1:Y       1:Y        *1      0
61       1:Y       1:Y        *0.995  0.005
62       1:Y       1:Y        *0.999  0.001
63       1:Y       2:N      +   0.362 *0.638
64       1:Y       1:Y        *0.997  0.003
65       1:Y       1:Y        *0.999  0.001
66       1:Y       1:Y        *0.999  0.001
67       2:N       1:Y      +  *0.996  0.004
68       1:Y       1:Y        *1      0
```

```
=== Evaluation on test split ===
=== Summary ===

Correctly Classified Instances          54               79.4118 %
Incorrectly Classified Instances        14               20.5882 %
Kappa statistic                          0.5657
K&B Relative Info Score               3831.5911 %
K&B Information Score                   33.7675 bits      0.4966
bits/instance
Class complexity | order 0              64.3284 bits      0.946
bits/instance
Class complexity | scheme               53.0961 bits      0.7808
bits/instance
Complexity improvement     (Sf)         11.2323 bits      0.1652
bits/instance
Mean absolute error                      0.1995
Root mean squared error                  0.3904
Relative absolute error                 45.214  %
Root relative squared error             81.1934 %
Total Number of Instances               68

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   Class
  0.795     0.208      0.875     0.795     0.833    Y
  0.792     0.205      0.679     0.792     0.731    N

=== Confusion Matrix ===

  a  b   <-- classified as
 35  9 |  a = Y
  5 19 |  b = N
```

```
Scheme:       weka.classifiers.functions.MultilayerPerceptron -L 0.3 -M 0.2
-N 500 -V 0 -S 0 -E 20 -H a
Relation:     Image_Features
Instances:    136
Attributes:   23
              File_Name
              Malignant
              Lesion_Nr_Points
              Edge_Nr_Points
              Dir_Changes
              Rays_Sigma
              C_Total
              C_Mean
              C_Sigma
              CD_Total
              CD_Mean
              CD_Sigma
              X_Total
              X_Mean
              X_Sigma
              Y_Total
              Y_Mean
              Y_Sigma
              DC_EP
              TCD_TC
              Cmin
              Cmax
              EP_LP
Test mode:    split 60% train, remainder test
Class Y
    Input
    Node 0
Class N
    Input
    Node 1
Time taken to build model: 117.36 seconds
```

```
=== Predictions on test split ===

 inst#,     actual, predicted, error, probability distribution
     1        1:Y        1:Y         *0.971  0.029
     2        1:Y        1:Y         *1      0
     3        2:N        2:N          0.002 *0.998
     4        1:Y        1:Y         *0.997  0.003
     5        2:N        2:N          0.001 *0.999
     6        1:Y        1:Y         *0.997  0.003
     7        1:Y        2:N      +   0.196 *0.804
     8        1:Y        1:Y         *1      0
     9        2:N        2:N          0.083 *0.917
    10        1:Y        2:N      +   0.243 *0.757
    11        1:Y        1:Y         *1      0
    12        1:Y        1:Y         *0.99   0.01
    13        1:Y        1:Y         *0.999  0.001
    14        2:N        2:N          0.001 *0.999
    15        1:Y        1:Y         *0.997  0.003
    16        1:Y        2:N      +   0.346 *0.654
    17        2:N        2:N          0     *1
    18        2:N        2:N          0.001 *0.999
    19        1:Y        1:Y         *0.999  0.001
    20        2:N        1:Y      +  *0.801  0.199
    21        2:N        2:N          0.009 *0.991
    22        1:Y        1:Y         *1      0
    23        2:N        2:N          0.149 *0.851
    24        1:Y        2:N      +   0.005 *0.995
    25        1:Y        1:Y         *0.877  0.123
    26        1:Y        2:N      +   0.47   *0.53
    27        2:N        2:N          0.002 *0.998
    28        1:Y        2:N      +   0.363 *0.637
    29        2:N        2:N          0.001 *0.999
    30        1:Y        1:Y         *0.999  0.001
    31        1:Y        1:Y         *1      0
    32        2:N        2:N          0.002 *0.998
    33        2:N        2:N          0.002 *0.998
```

```
34       2:N       2:N          0.014 *0.986
35       2:N       2:N          0.456 *0.544
36       1:Y       1:Y         *1      0
37       1:Y       1:Y         *0.943  0.057
38       1:Y       1:Y         *0.998  0.002
39       2:N       2:N          0.122 *0.878
40       2:N       2:N          0.395 *0.605
41       1:Y       1:Y         *0.999  0.001
42       1:Y       1:Y         *1      0
43       1:Y       1:Y         *0.993  0.007
44       2:N       2:N          0.095 *0.905
45       2:N       2:N          0.229 *0.771
46       1:Y       1:Y         *1      0
47       1:Y       1:Y         *1      0
48       1:Y       1:Y         *0.998  0.002
49       1:Y       1:Y         *0.999  0.001
50       1:Y       2:N      +   0.112 *0.888
51       1:Y       1:Y         *0.992  0.008
52       1:Y       1:Y         *0.996  0.004
53       1:Y       1:Y         *0.992  0.008
54       2:N       1:Y      +  *0.985  0.015
55       1:Y       1:Y         *1      0
```

```
=== Evaluation on test split ===
=== Summary ===

Correctly Classified Instances          46               83.6364 %
Incorrectly Classified Instances         9               16.3636 %
Kappa statistic                          0.6644
K&B Relative Info Score               3713.9135 %
K&B Information Score                   32.7849 bits        0.5961
bits/instance
Class complexity | order 0              52.7207 bits        0.9586
bits/instance
Class complexity | scheme               30.7654 bits        0.5594
bits/instance
Complexity improvement       (Sf)       21.9553 bits        0.3992
bits/instance
Mean absolute error                      0.1615
Root mean squared error                  0.3361
Relative absolute error                 36.2209 %
Root relative squared error             69.2915 %
Total Number of Instances               55

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   Class
  0.8       0.1        0.933     0.8       0.862    Y
  0.9       0.2        0.72      0.9       0.8      N

=== Confusion Matrix ===

  a  b   <-- classified as
 28  7 |  a = Y
  2 18 |  b = N
```

```
Scheme:       weka.classifiers.functions.MultilayerPerceptron -L 0.3 -M 0.2
-N 500 -V 0 -S 0 -E 20 -H a
Relation:     Image_Features
Instances:    136
Attributes:   23
              File_Name
              Malignant
              Lesion_Nr_Points
              Edge_Nr_Points
              Dir_Changes
              Rays_Sigma
              C_Total
              C_Mean
              C_Sigma
              CD_Total
              CD_Mean
              CD_Sigma
              X_Total
              X_Mean
              X_Sigma
              Y_Total
              Y_Mean
              Y_Sigma
              DC_EP
              TCD_TC
              Cmin
              Cmax
              EP_LP
Test mode:    split 70% train, remainder test
Class Y
    Input
    Node 0
Class N
    Input
    Node 1
Time taken to build model: 123.57 seconds
```

```
=== Predictions on test split ===

 inst#,     actual, predicted, error, probability distribution
     1        1:Y        1:Y         *0.997  0.003
     2        1:Y        1:Y         *0.517  0.483
     3        2:N        2:N          0.001 *0.999
     4        2:N        2:N          0.002 *0.998
     5        1:Y        1:Y         *0.998  0.002
     6        2:N        1:Y      +  *0.889  0.111
     7        2:N        2:N          0.018 *0.982
     8        1:Y        1:Y         *1      0
     9        2:N        2:N          0.197 *0.803
    10        1:Y        2:N      +   0.015 *0.985
    11        1:Y        1:Y         *0.948  0.052
    12        1:Y        1:Y         *0.539  0.461
    13        2:N        2:N          0.004 *0.996
    14        1:Y        2:N      +   0.425 *0.575
    15        2:N        2:N          0.002 *0.998
    16        1:Y        1:Y         *0.999  0.001
    17        1:Y        1:Y         *1      0
    18        2:N        2:N          0.007 *0.993
    19        2:N        2:N          0.002 *0.998
    20        2:N        2:N          0.034 *0.966
    21        2:N        2:N          0.414 *0.586
    22        1:Y        1:Y         *0.999  0.001
    23        1:Y        1:Y         *0.974  0.026
    24        1:Y        1:Y         *0.998  0.002
    25        2:N        2:N          0.143 *0.857
    26        2:N        1:Y      +  *0.554  0.446
    27        1:Y        1:Y         *0.999  0.001
    28        1:Y        1:Y         *1      0
    29        1:Y        1:Y         *0.994  0.006
    30        2:N        2:N          0.119 *0.881
    31        2:N        2:N          0.31  *0.69
    32        1:Y        1:Y         *1      0
    33        1:Y        1:Y         *1      0
```

```
    34        1:Y        1:Y          *0.999  0.001
    35        1:Y        1:Y          *0.999  0.001
    36        1:Y        2:N      +    0.341 *0.659
    37        1:Y        1:Y          *0.996  0.004
    38        1:Y        1:Y          *0.997  0.003
    39        1:Y        1:Y          *0.997  0.003
    40        2:N        1:Y      +   *0.971  0.029
    41        1:Y        1:Y          *1      0

=== Evaluation on test split ===
=== Summary ===

Correctly Classified Instances          35               85.3659 %
Incorrectly Classified Instances         6               14.6341 %
Kappa statistic                          0.6925
K&B Relative Info Score               2869.5045 %
K&B Information Score                            25.2524 bits        0.6159
bits/instance
Class complexity | order 0                       40.6822 bits        0.9922
bits/instance
Class complexity | scheme                         22.4264 bits         0.547
bits/instance
Complexity improvement       (Sf)                18.2558 bits        0.4453
bits/instance
Mean absolute error                      0.1691
Root mean squared error                  0.3337
Relative absolute error                 37.1044 %
Root relative squared error             67.2514 %
Total Number of Instances               41
```

```
=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   Class
  0.88      0.188      0.88      0.88      0.88     Y
  0.813     0.12       0.813     0.813     0.813    N

=== Confusion Matrix ===

  a  b   <-- classified as
 22  3 |  a = Y
  3 13 |  b = N
```

```
Scheme:       weka.classifiers.functions.MultilayerPerceptron -L 0.3 -M 0.2
-N 500 -V 0 -S 0 -E 20 -H a
Relation:     Image_Features
Instances:    136
Attributes:   23
              File_Name
              Malignant
              Lesion_Nr_Points
              Edge_Nr_Points
              Dir_Changes
              Rays_Sigma
              C_Total
              C_Mean
              C_Sigma
              CD_Total
              CD_Mean
              CD_Sigma
              X_Total
              X_Mean
              X_Sigma
              Y_Total
              Y_Mean
              Y_Sigma
              DC_EP
              TCD_TC
              Cmin
              Cmax
              EP_LP
Test mode:    split 80% train, remainder test
Class Y
    Input
    Node 0
Class N
    Input
    Node 1
Time taken to build model: 118.58 seconds
```

```
=== Predictions on test split ===

 inst#,     actual, predicted, error, probability distribution
      1        1:Y        1:Y         *0.518  0.482
      2        2:N        2:N          0.002 *0.998
      3        1:Y        1:Y         *0.999  0.001
      4        1:Y        1:Y         *1      0
      5        2:N        2:N          0.008 *0.992
      6        2:N        2:N          0.003 *0.997
      7        2:N        2:N          0.063 *0.937
      8        2:N        2:N          0.391 *0.609
      9        1:Y        1:Y         *1      0
     10        1:Y        1:Y         *0.977  0.023
     11        1:Y        1:Y         *0.999  0.001
     12        2:N        2:N          0.155 *0.845
     13        2:N        1:Y      +  *0.664  0.336
     14        1:Y        1:Y         *1      0
     15        1:Y        1:Y         *1      0
     16        1:Y        1:Y         *0.998  0.002
     17        2:N        2:N          0.399 *0.601
     18        2:N        2:N          0.318 *0.682
     19        1:Y        1:Y         *1      0
     20        1:Y        1:Y         *1      0
     21        1:Y        1:Y         *1      0
     22        1:Y        1:Y         *1      0
     23        1:Y        1:Y         *0.668  0.332
     24        1:Y        1:Y         *0.997  0.003
     25        1:Y        1:Y         *0.999  0.001
     26        1:Y        1:Y         *0.998  0.002
     27        2:N        1:Y      +  *0.992  0.008
     28        1:Y        1:Y         *1      0
```

```
=== Evaluation on test split ===
=== Summary ===

Correctly Classified Instances          26               92.8571 %
Incorrectly Classified Instances         2                7.1429 %
Kappa statistic                          0.8372
K&B Relative Info Score               2053.374  %
K&B Information Score                   18.5295 bits        0.6618
bits/instance
Class complexity | order 0              26.4665 bits        0.9452
bits/instance
Class complexity | scheme               12.458  bits        0.4449
bits/instance
Complexity improvement       (Sf)       14.0085 bits        0.5003
bits/instance
Mean absolute error                      0.1372
Root mean squared error                  0.2809
Relative absolute error                 30.6199 %
Root relative squared error             58.4236 %
Total Number of Instances               28

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   Class
  1         0.2        0.9       1         0.947    Y
  0.8       0          1         0.8       0.889    N

=== Confusion Matrix ===

  a  b   <-- classified as
 18  0 |  a = Y
  2  8 |  b = N
```

```
Scheme:       weka.classifiers.rules.ZeroR
Relation:     Image_Features
Instances:    136
Attributes:   23
              File_Name
              Malignant
              Lesion_Nr_Points
              Edge_Nr_Points
              Dir_Changes
              Rays_Sigma
              C_Total
              C_Mean
              C_Sigma
              CD_Total
              CD_Mean
              CD_Sigma
              X_Total
              X_Mean
              X_Sigma
              Y_Total
              Y_Mean
              Y_Sigma
              DC_EP
              TCD_TC
              Cmin
              Cmax
              EP_LP
Test mode:    split 50% train, remainder test

=== Classifier model (full training set) ===

ZeroR predicts class value: Y

Time taken to build model: 0 seconds
```

```
=== Predictions on test split ===

 inst#,     actual, predicted, error, probability distribution
     1        2:N        1:Y      +  *0.7    0.3
     2        1:Y        1:Y         *0.7    0.3
     3        1:Y        1:Y         *0.7    0.3
     4        2:N        1:Y      +  *0.7    0.3
     5        2:N        1:Y      +  *0.7    0.3
     6        1:Y        1:Y         *0.7    0.3
     7        1:Y        1:Y         *0.7    0.3
     8        1:Y        1:Y         *0.7    0.3
     9        1:Y        1:Y         *0.7    0.3
    10        1:Y        1:Y         *0.7    0.3
    11        2:N        1:Y      +  *0.7    0.3
    12        2:N        1:Y      +  *0.7    0.3
    13        1:Y        1:Y         *0.7    0.3
    14        1:Y        1:Y         *0.7    0.3
    15        2:N        1:Y      +  *0.7    0.3
    16        1:Y        1:Y         *0.7    0.3
    17        1:Y        1:Y         *0.7    0.3
    18        1:Y        1:Y         *0.7    0.3
    19        2:N        1:Y      +  *0.7    0.3
    20        2:N        1:Y      +  *0.7    0.3
    21        1:Y        1:Y         *0.7    0.3
    22        1:Y        1:Y         *0.7    0.3
    23        2:N        1:Y      +  *0.7    0.3
    24        2:N        1:Y      +  *0.7    0.3
    25        1:Y        1:Y         *0.7    0.3
    26        1:Y        1:Y         *0.7    0.3
    27        2:N        1:Y      +  *0.7    0.3
    28        1:Y        1:Y         *0.7    0.3
    29        1:Y        1:Y         *0.7    0.3
    30        2:N        1:Y      +  *0.7    0.3
    31        1:Y        1:Y         *0.7    0.3
    32        1:Y        1:Y         *0.7    0.3
    33        1:Y        1:Y         *0.7    0.3
```

34	1:Y	1:Y		*0.7	0.3
35	1:Y	1:Y		*0.7	0.3
36	1:Y	1:Y		*0.7	0.3
37	2:N	1:Y	+	*0.7	0.3
38	2:N	1:Y	+	*0.7	0.3
39	1:Y	1:Y		*0.7	0.3
40	2:N	1:Y	+	*0.7	0.3
41	1:Y	1:Y		*0.7	0.3
42	1:Y	1:Y		*0.7	0.3
43	1:Y	1:Y		*0.7	0.3
44	2:N	1:Y	+	*0.7	0.3
45	1:Y	1:Y		*0.7	0.3
46	1:Y	1:Y		*0.7	0.3
47	1:Y	1:Y		*0.7	0.3
48	2:N	1:Y	+	*0.7	0.3
49	1:Y	1:Y		*0.7	0.3
50	1:Y	1:Y		*0.7	0.3
51	2:N	1:Y	+	*0.7	0.3
52	1:Y	1:Y		*0.7	0.3
53	2:N	1:Y	+	*0.7	0.3
54	1:Y	1:Y		*0.7	0.3
55	2:N	1:Y	+	*0.7	0.3
56	1:Y	1:Y		*0.7	0.3
57	2:N	1:Y	+	*0.7	0.3
58	1:Y	1:Y		*0.7	0.3
59	1:Y	1:Y		*0.7	0.3
60	2:N	1:Y	+	*0.7	0.3
61	1:Y	1:Y		*0.7	0.3
62	2:N	1:Y	+	*0.7	0.3
63	2:N	1:Y	+	*0.7	0.3
64	1:Y	1:Y		*0.7	0.3
65	1:Y	1:Y		*0.7	0.3
66	1:Y	1:Y		*0.7	0.3
67	1:Y	1:Y		*0.7	0.3
68	1:Y	1:Y		*0.7	0.3

```
=== Evaluation on test split ===
=== Summary ===

Correctly Classified Instances          44               64.7059 %
Incorrectly Classified Instances        24               35.2941 %
Kappa statistic                          0
K&B Relative Info Score                  0      %
K&B  Information  Score                            0         bits         0
bits/instance
Class  complexity  |  order  0                     64.3284  bits         0.946
bits/instance
Class  complexity  |  scheme                      64.3284  bits         0.946
bits/instance
Complexity  improvement        (Sf)                0          bits          0
bits/instance
Mean absolute error                      0.4412
Root mean squared error                  0.4808
Relative absolute error                100      %
Root relative squared error            100      %
Total Number of Instances               68

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   Class
  1         1          0.647     1         0.786    Y
  0         0          0         0         0        N

=== Confusion Matrix ===

  a  b   <-- classified as
 44  0 |  a = Y
 24  0 |  b = N
```

```
Scheme:       weka.classifiers.rules.ZeroR
Relation:     Image_Features
Instances:    136
Attributes:   23
              File_Name
              Malignant
              Lesion_Nr_Points
              Edge_Nr_Points
              Dir_Changes
              Rays_Sigma
              C_Total
              C_Mean
              C_Sigma
              CD_Total
              CD_Mean
              CD_Sigma
              X_Total
              X_Mean
              X_Sigma
              Y_Total
              Y_Mean
              Y_Sigma
              DC_EP
              TCD_TC
              Cmin
              Cmax
              EP_LP
Test mode:    split 60% train, remainder test

=== Classifier model (full training set) ===

ZeroR predicts class value: Y

Time taken to build model: 0 seconds
```

```
=== Predictions on test split ===

 inst#,     actual, predicted, error, probability distribution
     1        1:Y        1:Y          *0.687  0.313
     2        2:N        1:Y      +   *0.687  0.313
     3        1:Y        1:Y          *0.687  0.313
     4        1:Y        1:Y          *0.687  0.313
     5        1:Y        1:Y          *0.687  0.313
     6        2:N        1:Y      +   *0.687  0.313
     7        2:N        1:Y      +   *0.687  0.313
     8        1:Y        1:Y          *0.687  0.313
     9        1:Y        1:Y          *0.687  0.313
    10        2:N        1:Y      +   *0.687  0.313
    11        2:N        1:Y      +   *0.687  0.313
    12        1:Y        1:Y          *0.687  0.313
    13        1:Y        1:Y          *0.687  0.313
    14        2:N        1:Y      +   *0.687  0.313
    15        1:Y        1:Y          *0.687  0.313
    16        1:Y        1:Y          *0.687  0.313
    17        2:N        1:Y      +   *0.687  0.313
    18        1:Y        1:Y          *0.687  0.313
    19        1:Y        1:Y          *0.687  0.313
    20        1:Y        1:Y          *0.687  0.313
    21        1:Y        1:Y          *0.687  0.313
    22        1:Y        1:Y          *0.687  0.313
    23        1:Y        1:Y          *0.687  0.313
    24        2:N        1:Y      +   *0.687  0.313
    25        2:N        1:Y      +   *0.687  0.313
    26        1:Y        1:Y          *0.687  0.313
    27        2:N        1:Y      +   *0.687  0.313
    28        1:Y        1:Y          *0.687  0.313
    29        1:Y        1:Y          *0.687  0.313
    30        1:Y        1:Y          *0.687  0.313
    31        2:N        1:Y      +   *0.687  0.313
    32        1:Y        1:Y          *0.687  0.313
    33        1:Y        1:Y          *0.687  0.313
```

34	1:Y	1:Y		*0.687	0.313
35	2:N	1:Y	+	*0.687	0.313
36	1:Y	1:Y		*0.687	0.313
37	1:Y	1:Y		*0.687	0.313
38	2:N	1:Y	+	*0.687	0.313
39	1:Y	1:Y		*0.687	0.313
40	2:N	1:Y	+	*0.687	0.313
41	1:Y	1:Y		*0.687	0.313
42	2:N	1:Y	+	*0.687	0.313
43	1:Y	1:Y		*0.687	0.313
44	2:N	1:Y	+	*0.687	0.313
45	1:Y	1:Y		*0.687	0.313
46	1:Y	1:Y		*0.687	0.313
47	2:N	1:Y	+	*0.687	0.313
48	1:Y	1:Y		*0.687	0.313
49	2:N	1:Y	+	*0.687	0.313
50	2:N	1:Y	+	*0.687	0.313
51	1:Y	1:Y		*0.687	0.313
52	1:Y	1:Y		*0.687	0.313
53	1:Y	1:Y		*0.687	0.313
54	1:Y	1:Y		*0.687	0.313
55	1:Y	1:Y		*0.687	0.313

```
=== Evaluation on test split ===
=== Summary ===

Correctly Classified Instances          36               65.4545 %
Incorrectly Classified Instances        19               34.5455 %
Kappa statistic                          0
K&B Relative Info Score                  0      %
K&B  Information  Score                            0         bits         0
bits/instance
Class  complexity  |  order  0                    51.3348  bits         0.9334
bits/instance
Class  complexity  |  scheme                     51.3348  bits         0.9334
bits/instance
Complexity  improvement        (Sf)                 0          bits          0
bits/instance
Mean absolute error                      0.4423
Root mean squared error                  0.4766
Relative absolute error                100      %
Root relative squared error            100      %
Total Number of Instances               55

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   Class
  1         1          0.655     1         0.791    Y
  0         0          0         0         0        N

=== Confusion Matrix ===

  a  b   <-- classified as
 36  0 |  a = Y
 19  0 |  b = N
```

```
Scheme:       weka.classifiers.rules.ZeroR
Relation:     Image_Features
Instances:    136
Attributes:   23
              File_Name
              Malignant
              Lesion_Nr_Points
              Edge_Nr_Points
              Dir_Changes
              Rays_Sigma
              C_Total
              C_Mean
              C_Sigma
              CD_Total
              CD_Mean
              CD_Sigma
              X_Total
              X_Mean
              X_Sigma
              Y_Total
              Y_Mean
              Y_Sigma
              DC_EP
              TCD_TC
              Cmin
              Cmax
              EP_LP
Test mode:    split 70% train, remainder test

=== Classifier model (full training set) ===

ZeroR predicts class value: Y

Time taken to build model: 0 seconds
```

```
=== Predictions on test split ===

 inst#,     actual, predicted, error, probability distribution
     1        1:Y        1:Y          *0.67   0.33
     2        1:Y        1:Y          *0.67   0.33
     3        2:N        1:Y      +   *0.67   0.33
     4        1:Y        1:Y          *0.67   0.33
     5        1:Y        1:Y          *0.67   0.33
     6        1:Y        1:Y          *0.67   0.33
     7        1:Y        1:Y          *0.67   0.33
     8        1:Y        1:Y          *0.67   0.33
     9        1:Y        1:Y          *0.67   0.33
    10        2:N        1:Y      +   *0.67   0.33
    11        2:N        1:Y      +   *0.67   0.33
    12        1:Y        1:Y          *0.67   0.33
    13        2:N        1:Y      +   *0.67   0.33
    14        1:Y        1:Y          *0.67   0.33
    15        1:Y        1:Y          *0.67   0.33
    16        1:Y        1:Y          *0.67   0.33
    17        2:N        1:Y      +   *0.67   0.33
    18        1:Y        1:Y          *0.67   0.33
    19        1:Y        1:Y          *0.67   0.33
    20        1:Y        1:Y          *0.67   0.33
    21        2:N        1:Y      +   *0.67   0.33
    22        1:Y        1:Y          *0.67   0.33
    23        1:Y        1:Y          *0.67   0.33
    24        2:N        1:Y      +   *0.67   0.33
    25        1:Y        1:Y          *0.67   0.33
    26        2:N        1:Y      +   *0.67   0.33
    27        1:Y        1:Y          *0.67   0.33
    28        2:N        1:Y      +   *0.67   0.33
    29        1:Y        1:Y          *0.67   0.33
    30        2:N        1:Y      +   *0.67   0.33
    31        1:Y        1:Y          *0.67   0.33
    32        1:Y        1:Y          *0.67   0.33
    33        2:N        1:Y      +   *0.67   0.33
```

34	1:Y	1:Y		*0.67	0.33
35	2:N	1:Y	+	*0.67	0.33
36	2:N	1:Y	+	*0.67	0.33
37	1:Y	1:Y		*0.67	0.33
38	1:Y	1:Y		*0.67	0.33
39	1:Y	1:Y		*0.67	0.33
40	1:Y	1:Y		*0.67	0.33
41	1:Y	1:Y		*0.67	0.33

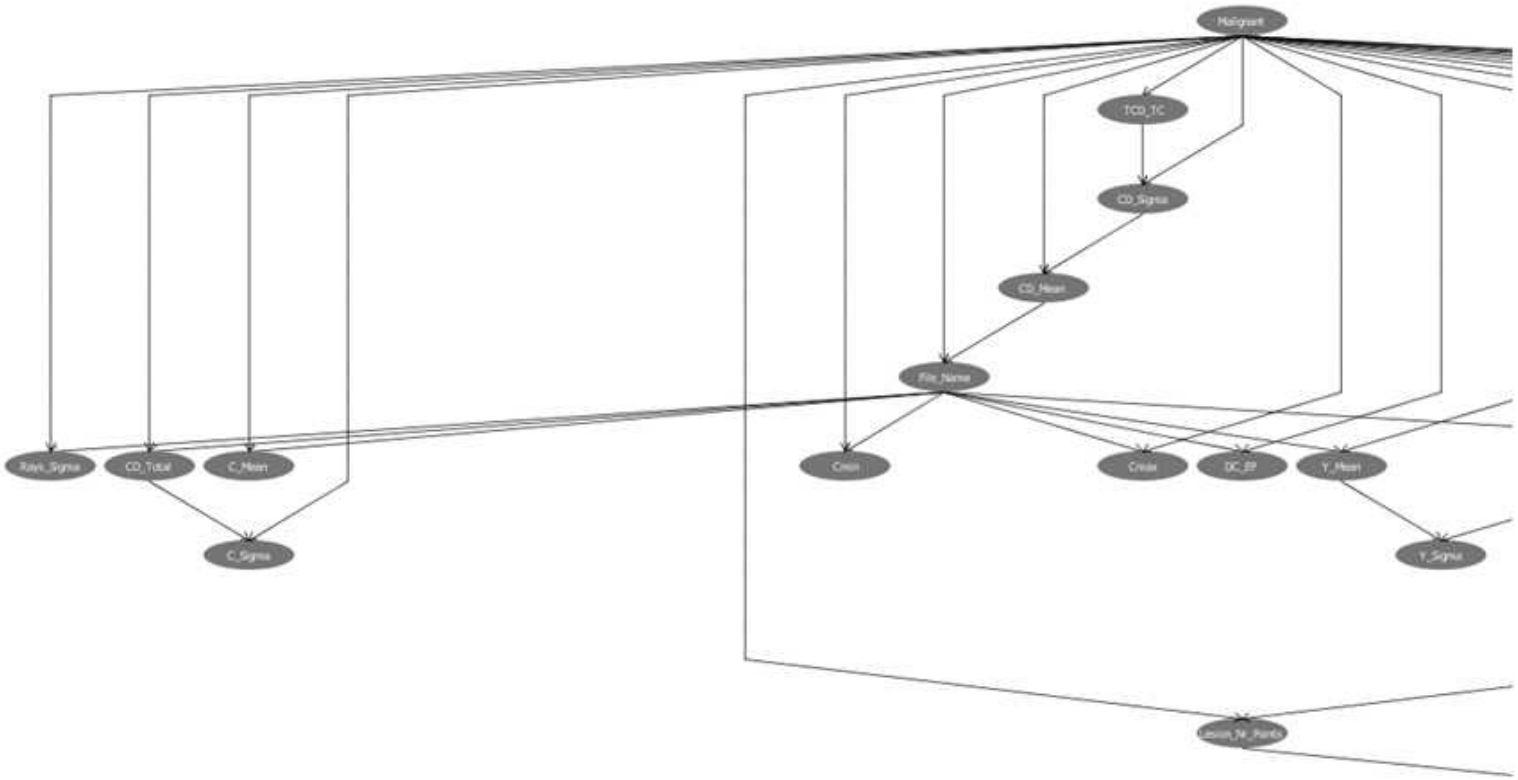
```
=== Evaluation on test split ===
=== Summary ===

Correctly Classified Instances          28               68.2927 %
Incorrectly Classified Instances        13               31.7073 %
Kappa statistic                          0
K&B Relative Info Score                  0       %
K&B Information Score                                0          bits          0
bits/instance
Class complexity | order 0                        36.9701  bits         0.9017
bits/instance
Class complexity | scheme                         36.9701  bits         0.9017
bits/instance
Complexity improvement        (Sf)                  0          bits          0
bits/instance
Mean absolute error                      0.4378
Root mean squared error                  0.4655
Relative absolute error                100       %
Root relative squared error            100       %
Total Number of Instances               41

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   Class
  1         1          0.683     1         0.812    Y
  0         0          0         0         0        N


=== Confusion Matrix ===

  a  b   <-- classified as
 28  0 |  a = Y
 13  0 |  b = N
```

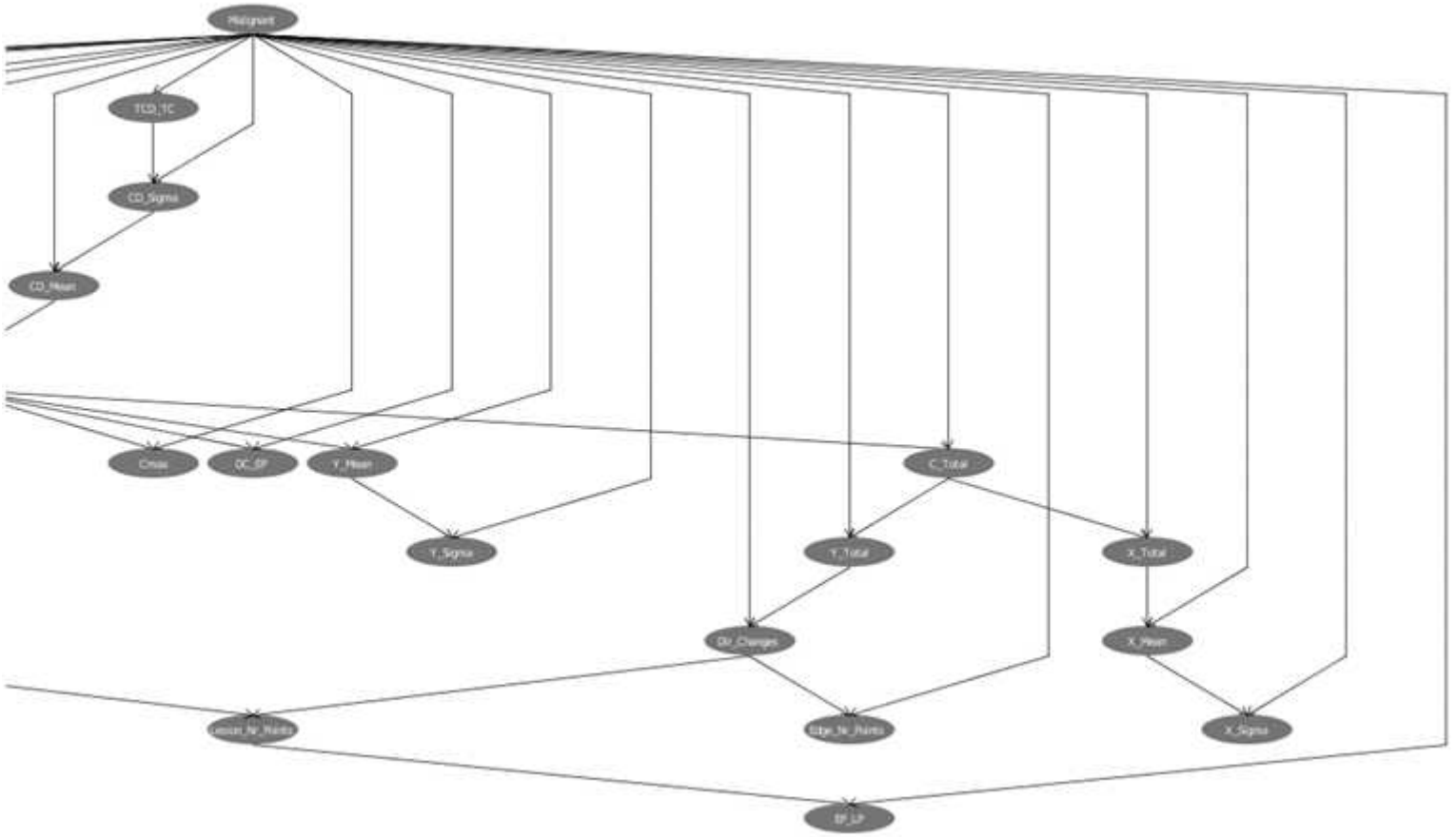
```
Scheme:       weka.classifiers.rules.ZeroR
Relation:     Image_Features
Instances:    136
Attributes:   23
              File_Name
              Malignant
              Lesion_Nr_Points
              Edge_Nr_Points
              Dir_Changes
              Rays_Sigma
              C_Total
              C_Mean
              C_Sigma
              CD_Total
              CD_Mean
              CD_Sigma
              X_Total
              X_Mean
              X_Sigma
              Y_Total
              Y_Mean
              Y_Sigma
              DC_EP
              TCD_TC
              Cmin
              Cmax
              EP_LP
Test mode:    split 80% train, remainder test

=== Classifier model (full training set) ===

ZeroR predicts class value: Y

Time taken to build model: 0 seconds
```

```
=== Predictions on test split ===

 inst#,     actual, predicted, error, probability distribution
     1        1:Y        1:Y          *0.673  0.327
     2        1:Y        1:Y          *0.673  0.327
     3        1:Y        1:Y          *0.673  0.327
     4        2:N        1:Y      +   *0.673  0.327
     5        1:Y        1:Y          *0.673  0.327
     6        1:Y        1:Y          *0.673  0.327
     7        1:Y        1:Y          *0.673  0.327
     8        2:N        1:Y      +   *0.673  0.327
     9        1:Y        1:Y          *0.673  0.327
    10        1:Y        1:Y          *0.673  0.327
    11        2:N        1:Y      +   *0.673  0.327
    12        1:Y        1:Y          *0.673  0.327
    13        2:N        1:Y      +   *0.673  0.327
    14        1:Y        1:Y          *0.673  0.327
    15        2:N        1:Y      +   *0.673  0.327
    16        1:Y        1:Y          *0.673  0.327
    17        2:N        1:Y      +   *0.673  0.327
    18        1:Y        1:Y          *0.673  0.327
    19        1:Y        1:Y          *0.673  0.327
    20        2:N        1:Y      +   *0.673  0.327
    21        1:Y        1:Y          *0.673  0.327
    22        2:N        1:Y      +   *0.673  0.327
    23        2:N        1:Y      +   *0.673  0.327
    24        1:Y        1:Y          *0.673  0.327
    25        1:Y        1:Y          *0.673  0.327
    26        1:Y        1:Y          *0.673  0.327
    27        1:Y        1:Y          *0.673  0.327
    28        1:Y        1:Y          *0.673  0.327
```

```
=== Evaluation on test split ===
=== Summary ===

Correctly Classified Instances          19               67.8571 %
Incorrectly Classified Instances         9               32.1429 %
Kappa statistic                          0
K&B Relative Info Score                  0      %
K&B  Information  Score                          0         bits         0
bits/instance
Class  complexity  |  order  0                    25.3691  bits         0.906
bits/instance
Class  complexity  |  scheme                     25.3691  bits         0.906
bits/instance
Complexity  improvement        (Sf)               0         bits         0
bits/instance
Mean absolute error                      0.4383
Root mean squared error                  0.4671
Relative absolute error                100      %
Root relative squared error            100      %
Total Number of Instances               28

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   Class
  1         1          0.679     1         0.809    Y
  0         0          0         0         0        N

=== Confusion Matrix ===

  a  b   <-- classified as
 19  0 |  a = Y
  9  0 |  b = N
```

8.7 Appendix G - Tree Augmented Naïve Bayes (TAN) tree


Malignant
TCD_TC
CD_Sigma
CD_Mean
File_Name
Ravir_Sigma
CD_Total
C_Mean
C_Sigma
Cmin
Cmax
DC_EP
Y_Mean
Y_Sigma


Malignant
TCD_TC
CD_Sigma
CD_Mean
Cmax
DC_EP
Y_Mean
C_Total
Y_Sigma
Y_Total
X_Total
Dir_Changes
X_Mean
X_Sigma
EP_LP

8.8 Appendix H – Process Flowchart