Public University of Navarre

Ultrasound Image Discrimination Between Benign and Malignant Adnexal Masses

Biomedical Engineering
Master Thesis

Verónica Aramendía Vidaurreta

Supervisors:
Rafael Cabeza and Arantxa Villanueva

Pamplona, February 2015

Abstract

This thesis deals with the discrimination between benign and malignant adnexal masses through ultrasound images. This task represents one of the most challenging problems in gynecological practice. Benign adnexal masses should be treated by minimally invasive surgery whereas patients with questionable adnexal masses should be referred for primary surgery. An accurate diagnosis is crucial in order to establish the optimal management for these patients. Therefore, it is essential for the specialist to have as many tools as possible in order to distinguish between benign and malignant adnexal masses. The diagnostic techniques that are being used involve $2D$ images and $3D$ ultrasound volumes. The basic outline of such system is the following. The first step consists of pre-processing both the images and volumes. Then, a set of characteristics is extracted. Finally, these characteristics will be used as an input to a classification system. The main goal of this thesis is make this system to be part of the daily clinical practice in order to validate its viability as an aid to diagnosis.

Nomenclature

ANN	Artificial Neural Networks
B-Mode	Brightness-Mode
Cd	Gray level Co-ocurrence Matrix
CV	Cross Validation
CAD	Computer Aided Diagnostic
CLT	Central Limit Theorem
E	Shannon Entropy
E3	Edge Texture Mask 3x3
EK	Kapur's Entropy
FD	Fractal Dimension
FP	False Positive
FN	False Negative
FOV	Field Of View
g_c	Center Pixel
gf	Gabor filter
gI	Gabor representation of an image
gk	Maximum gray level
gl	Minimum gray level
G	Gray levels
GLCM	Gray Level Co-occurrence Matrix
GW	Gabor Wavelet
h	Histogram
H	Normalized Histogram
hu	Hu Moment
I	Image
IM	Invariant Moments
k	Gabor Orientations
K	Cross Validation Divisions
l	Gabor Scales
L3	Level Texture Mask 3x3
LBP	Local Binary Pattern
LTE	Laws Texture Energy
MSE	Mean Squared Error
N	Number of pieces
PCA	Principal Component Analysis
P	Pixel Neighbors
Pd	Probability matrix: GLCM Normalized
r	Structure size
R	Radius
ROI	Region Of Interest
s	Scale factor
si	Silhouette value
S3	Spot Texture Mask 3x3

ST	Student Test
TE	Texture Energy Map
TI	Texture Image
TP	True Positive
TN	True Negative
US	Ultrasound
SA	Semi Automated
μ_{pq}	Central moment order (p+q)
η_{pq}	Normalized central moment order (p+q)

Acknowledgments

I would like to thank Rafael Cabeza and Arantxa Villanueva as my supervisors through the time it took me to complete this research. First of all, for the opportunity of starting with this interesting topic, but also for their contribution with new ideas during the process and their good-natured support. I must acknowledge as well Javier Navallas and Fermin Mallor, for their knowledge in neural networks and statistical methods. My sincere thanks also goes to Juan Luis Alcazar for offering us the whole image data base and for teaching us some of the medical aspects needed for the discrimination.

I would like to extend a warm appreciation to those family and friends who have been encouraging along the way for their support and patience. Thank you for being there in all these moments.

Contents

Contents v

List of Figures vii

List of Tables ix

1 Introduction 1
- 1.1 Adnexal Masses . . . 1
- 1.2 3D Ultrasound Examination . . . 2
  - 1.2.1 Ultrasound Scanning . . . 2
  - 1.2.2 Ultrasound Transducers . . . 3
  - 1.2.3 Ultrasound Presentation Modes . . . 4
- 1.3 Objective of the Thesis . . . 5
- 1.4 Structure of the Thesis . . . 6

2 Database 7
- 2.1 Classification Types . . . 8

3 Proposed System 9
- 3.1 Preprocessing Method . . . 9
  - 3.1.1 Semi-Automated Image Method . . . 10
  - 3.1.2 Non-Automated Method or ROI Method . . . 10
- 3.2 Diagram . . . 11

4 Extracted Features 12
- 4.1 Local Binary Pattern (LBP) . . . 12
  - 4.1.1 Histogram calculation . . . 12
  - 4.1.2 Average Power and Entropy calculation . . . 13
- 4.2 Fractal Dimension (FD) . . . 16
- 4.3 Entropy (E) . . . 18
- 4.4 Hu's Invariant Moments (IM) . . . 19
- 4.5 Gray Level Co-ocurrence Matrix (GLCM) . . . 20
- 4.6 Laws Texture Energy (LTE) . . . 22
- 4.7 Gabor Wavelet Transform (GW) . . . 24

5 Dimensionality Reduction 26
- 5.1 Feature Data . . . 26
- 5.2 Exploratory Data Analysis . . . 26
- 5.3 Feature selection . . . 31
  - 5.3.1 Student t-test of 2 populations . . . 31
  - 5.3.2 Results Semi-automated procedure . . . 31
  - 5.3.3 Results ROI procedure . . . 32
- 5.4 Feature extraction . . . 34

5.4.1 Principal Component Analysis (PCA) 34
5.4.2 Results Semi-automated procedure 35
5.4.3 Results ROI procedure 37
5.5 Hybrid Method: ST and PCA 39
5.6 Feature scaling 40
5.7 Feature vector Summary (FV) 40
5.7.1 FV Semiautomated Procedure 40
5.7.2 FV ROI Procedure 41

6 Classification System 42
6.1 Classifier Evaluation: K-fold cross validation (CV) 42
6.2 Artificial Neural Network Classifier (ANN) 43
6.2.1 Problems 45
6.2.2 Implementation 46
6.2.3 Network programming 52
6.2.4 Results 55
6.2.5 Summary 61

7 Conclusions 63

Bibliography 64

List of Figures

1.1 Front and Lateral View of the Adnexa of Uterus 1
1.2 Transabdominal and Transvaginal Ultrasound [18] 2
1.3 Transducer zones and resolutions 3
1.4 Transducer Beams: Sector, Linear and Curved respectively 4
1.5 Desired direction 5
1.6 B Mode 5

2.1 Front(A), Lateral(B) and Top view(C) from an Ultrasound Volume 7
2.2 Ultrasound Volume 7

3.1 Image plane from the volume 9
3.2 3D to 2D Volumes 10
3.3 ROI Image plane from the volume 10
3.4 Block Diagram System 11
3.5 Equal image height 11

4.1 LBP Example 13
4.2 LBP Scales: 81, 82 13
4.3 LBP Histogram 81 Example 14
4.4 LBP Histogram 81. Benign and Malignant. ROI 14
4.5 LBP Scales: 81, 162, 323 15
4.6 LBP Average Power Example 15
4.7 FD Example (Scale factor ($1/r$) and Number of boxes ($N(r)$)) 16
4.8 Box Counting Method Example 17
4.9 Differential Box Counting Method Example, [13] 17
4.10 Box Counting Method ROI Example 18
4.11 FD Example 18
4.12 E Example 19
4.13 IM Example 20
4.14 3 GLCM examples for a gray-tone image 21
4.15 GLCM calculation process: Original, Pseudocolor and GLCM 21
4.16 GLCM Correlation Example 22
4.17 LTE Masks 23
4.18 LTE Example 23
4.19 LTE Example 24
4.20 GW Example and GW filters 25

5.1 Diagram Dimension 28
5.2 Scatter plot SA and ROI 28
5.3 Scatter plot SA and ROI 29
5.4 Correlation Matrix for Semi-automated Procedure 30
5.5 Correlation Matrix for ROI Procedure 30

5.6 Histogram and Silhouette values for LBP Entropy 018 ROI Procedure 31
5.7 PCA Variability. Semi-automated Procedure 35
5.8 PCA 1 and 2 components. Semi-automated Procedure 36
5.9 PCA three components. Semi-automated Procedure 37
5.10 PCA Variability. ROI Procedure 37
5.11 PCA 1 and 2 components. ROI Procedure 38
5.12 PCA three components. ROI Procedure 39
5.13 Problems in ST and PCA 39
5.14 Solution in ST and PCA 40
5.15 Feature Vector 41

6.1 Cross validation example 42
6.2 Neural Network Relationship 43
6.3 Weights and Bias Example in Perceptron 44
6.4 Forward and Back propagation Example 44
6.5 Underfitting, Suitable and Overfitting Model Relationship 45
6.6 Learning Curve for Division 6 46
6.7 Implementation 47
6.8 Activation function examples 47
6.9 Two outputs network 48
6.10 Single output network 48
6.11 Data division 50
6.12 Data random divisions 50
6.13 Confusion Matrix 52
6.14 Diagram Network 53
6.15 Gradient descent 53
6.16 Performance (Early stopping) 54
6.17 Error histogram 55
6.18 PCA variabilities comparison 61
6.19 All methods comparison 62

List of Tables

2.1 39 Malignant Classification 8
2.2 106 Benign Classification 8

5.1 Student t-test significant results (p-value< 0.05). Semi-automated procedure . . . 32
5.2 Student t-test significant results (p-value< 0.05). ROI procedure 34

6.1 Results for the network $38 - 46 - 2$ 55
6.2 Results for ST Semi-automated Softmax 57
6.3 Results for PCA Semi-automated Softmax 57
6.4 Results for Hybrid Semi-automated Softmax 57
6.5 Results for ST Semi-automated Sigmoid 58
6.6 Results for PCA Semi-automated Sigmoid 58
6.7 Results for Hybrid Semi-automated Sigmoid 58
6.8 Results for ST ROI Softmax 59
6.9 Results for PCA ROI Softmax 59
6.10 Results for Hybrid ROI Softmax 59
6.11 Results for ST ROI Sigmoid 60
6.12 Results for PCA ROI Sigmoid 60
6.13 Results for Hybrid ROI Sigmoid 60
6.14 Summary average results for SA and ROI 62
6.15 Summary best division results for SA and ROI 62

Chapter 1

Introduction

1.1 Adnexal Masses

An adnexal mass is a lump in tissue of the adnexa of uterus, usually in the ovary or fallopian tube. Adnexal masses can be both benign or malignant. The adnexa of uterus or uterine appendages refers to those structures that are most closely related to the uterus both structurally and functionally, as we can see in Figure 1.1.

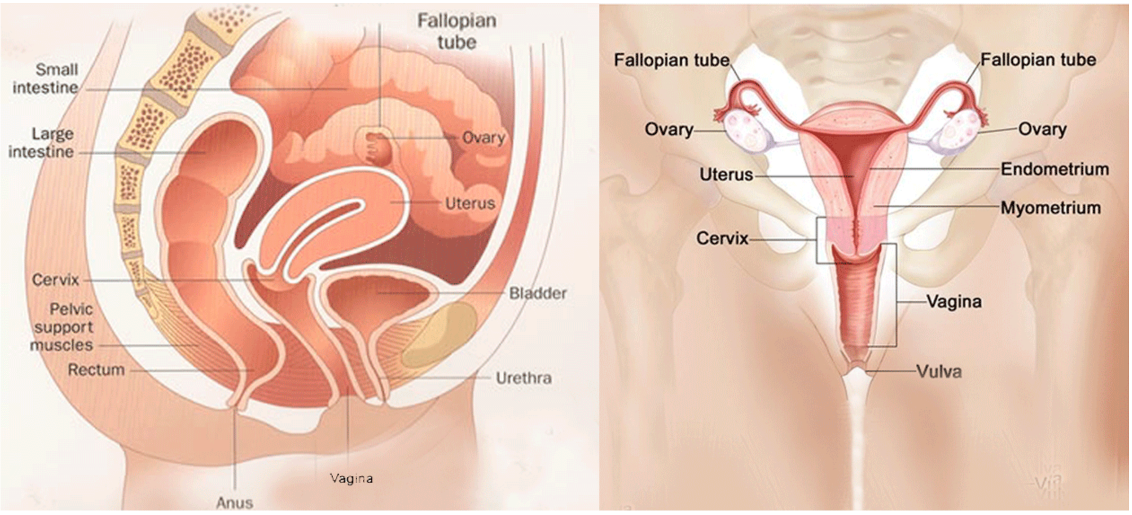


Figure 1.1: Front and Lateral View of the Adnexa of Uterus

Adnexal masses are frequently found in both symptomatic and asymptomatic women. Since ovaries produce physiologic cysts in menstruating women, the likelihood of a benign adnexal mass is higher in women of reproductive age. Malignant neoplasms are uncommon in younger women but become more frequent with increasing age and the incidence of malignancy rises. The overall risk of a primary ovarian neoplasm being malignant increases from 13% in premenopausal women to 45% following menopause [14]. In premenopausal women, physiologic follicular cysts and corpus luteum cysts are the most common adnexal masses.

Moreover, ovarian cancer is the most frequent cause of gynecological death. Therefore, it is essential for the specialist to have as many tools as possible in order to distinguish between malignant and benign adnexal masses and nowadays, it represents one of the biggest challenges in gynecological practice. $5-10\%$ of US women with a suspected adnexal mass will undergo surgery, but only $13-21\%$ will have a mass that is proven to be malignant (NHI Consensus Conference 1995) .

1.2 3D Ultrasound Examination

Ultrasound allows analysis in vivo of all the characteristics evaluated by surgeons and anatomical pathologists. The optimal ultrasound approach to characterize adnexal masses remains to be established. A diagnosis can be suspected on the basis of the morphological characteristics, such as its complexity, the presence of solid portions and irregularity. Many of this sonographic features are associated with a higher probability of malignancy. Furthermore, through a $3D$ ecography, a volume can be obtained and post-analyzed, which provides advantages, such as reconstructions or volume calculations.

1.2.1 Ultrasound Scanning

Technological advances have made possible to use different ways of ultrasound examinations such as transvaginal, transabdominal or transrectal scanning. In the following, transvaginal and transabdominal scanning will be explained in detail. In Figure 1.2, both examinations can be seen.

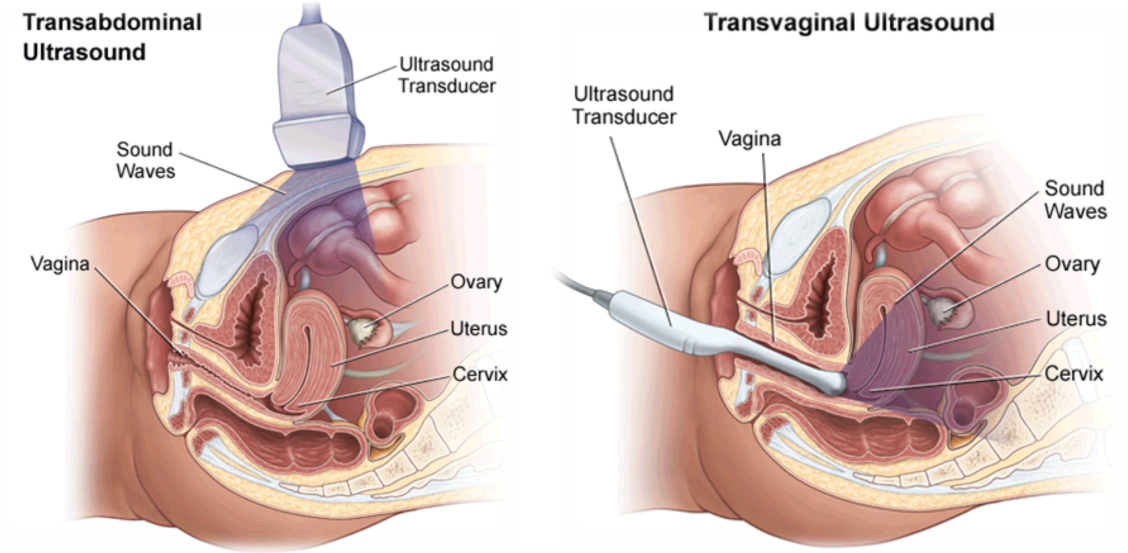


Figure 1.2: Transabdominal and Transvaginal Ultrasound [18]

Transvaginal scanning is an internal ultrasound which involves scanning with the ultrasound probe lying in the vagina. The ultrasound probe lies closer to the female pelvic organs and it operates at a higher frequency, so that more resolution can be achieved, specially in patients who are obese or in the early stages of pregnancy. However, some conditions limit transvaginal scanning: the integrity of the hymen, women's refusal to undergo an invasive imaging technique or the presence of cicatricial processes involving the vaginal walls that could make the transducer's movements painful or limit them.

Transabdominal ultrasound involves scanning through the lower abdomen. It usually provides an overview of the area rather than detailed images. It should be considered for use with transvaginal scanning in abdominopelvic neoformations that cannot be explored completely with transvaginal ultrasound and when a woman's condition does not allow endovaginal access.

From now on, we will focus on transvaginal examination. The database obtained will be based on it.

1.2.2 Ultrasound Transducers

An ultrasound transducer is a device that converts electrical energy into ultrasound energy and vice versa. It consists of one or more piezoelectric crystals or elements. The piezoelectric effect is exhibited by certain crystals that, in response to applied pressure, develop a voltage across opposite surfaces. This effect is used to produce an electrical signal in response to incident ultrasound waves.

An ultrasound transducer is designed to be maximally sensitive to ultrasound of a particular frequency, denominated resonance frequency of the transducer, which is mainly determined by the thickness of the piezoelectric crystal. Proper selection of the transducer frequency is an important concept for providing optimal image resolution in diagnostic. In Figure 1.3, it can be differentiated between lateral (C) and axial (B) resolution. Lateral resolution is the ability of the ultrasound system to display two objects side-by-side as separate structures. It is best in the focal zone, where the ultrasound beams are the narrowest and most concentrated. It depends on the distance between the individual crystals rather than the distance between the objects being viewed. Resolution diminishes in the far zone as the beam begins to diverge and is attenuated by tissue. Axial resolution relates to the ultrasound system's ability to differentiate objects in-line with the axis of the sound wave. It is dependent on the length of the sound impulse and the ultrasound frequency.

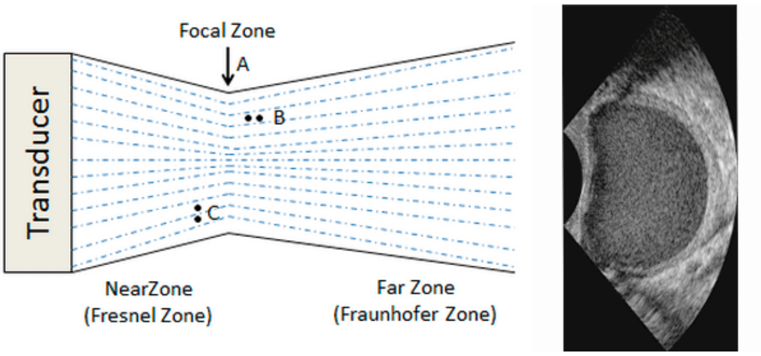


Figure 1.3: Transducer zones and resolutions

Therefore, high-frequency ultrasound waves generate images of high axial resolution, but are more attenuated than lower frequency waves for a given distance; thus, they are suitable for imaging mainly superficial structures. Conversely, low-frequency waves (long wavelength) offer images of lower resolution but can penetrate to deeper structures due to a lower degree of attenuation. This is the main limitation in ultrasound images. Lower frequencies are needed to study deep structures, but image resolution is automatically reduced due to the tissue attenuation.

The choice of the transducer will determine the shape and field of view (FOV) of the ultrasound image. It can be differentiated among:

- *Sector or Phased array.* They produce narrow images in the near-field but with a wide view in the far-field. Therefore, they are optimal for examining larger organs, for example those between the ribs.

- *Linear array.* They produce rectangular images. The width of the image is determined by the physical width of the transducer face. They often offer the best overall image quality and are preferred for examining anatomy in the near-field.

- *Curved or Convex array.* They are a cross between linear and sector transducers providing a broader view in the near-field while retaining a broad view in the far-field. The transducer face is wide and gently curved.

In this project, a $RIC5-9$ transvaginal wide curved transducer is used. As it is an internal ultrasound, it will operate at higher frequency and its bandwidth is between 4 and 9 Mhz. In this way, images with more resolution can be provided because it can normally be placed adjacent or very close to the uterus and ovaries structures.

This probe will be covered with a disposable protective sheath and a small amount of ultrasound gel is placed on the end of this probe in order to reduce the attenuation due to the sharp change among the structures, the air and the probe. This is important, because as we are dealing with sound waves, its velocity changes drastically from air to water or bone, for example. In Figure 1.4, the basic transducer formats can be seen. These are sector, linear and curved array respectively, together with an image of the ultrasound gel on the sheath of the probe.

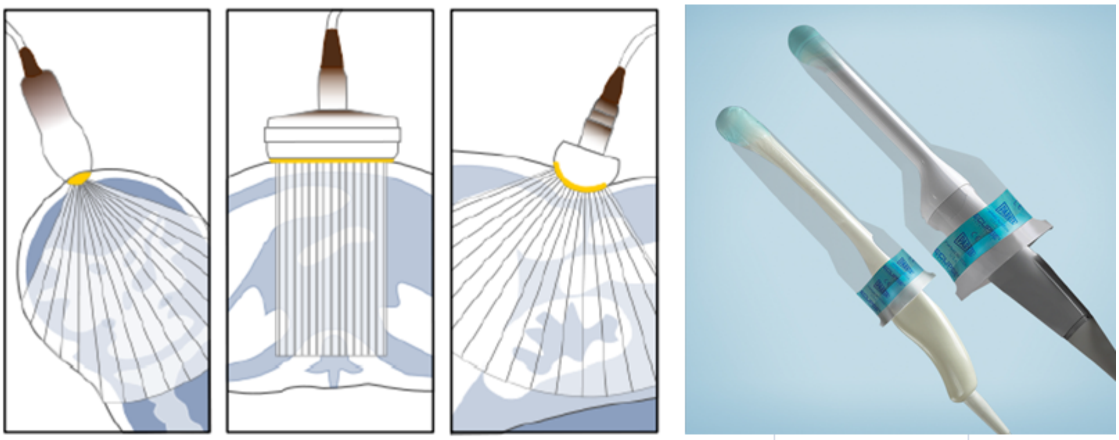


Figure 1.4: Transducer Beams: Sector, Linear and Curved respectively

As in Figure 1.5, in order to obtain the appropriate information from the ultrasound examination, the probe is pointed in the proper direction, pushed into the vaginal vault to the desired depth for maximal visualization, and rotated to alter the plane of transection. After the region of interest of the adnexa of uterus is found, a $3D$ ecography through the transvaginal scanning is made. An automatic method is used, from where the transducer makes a sweep over this region, creating the volume. The velocity and sweep angle is chosen by the expert. The lower speed and angle chosen, the more resolution is obtained. In general, it can be said that the quality of an automatic method is much better than the manual one.

1.2.3 Ultrasound Presentation Modes

This thesis will be focus on the B-Mode or Brightness-Mode representation. The basic principles of B-mode imaging involve transmitting small pulses of ultrasound echo from a transducer into the body.

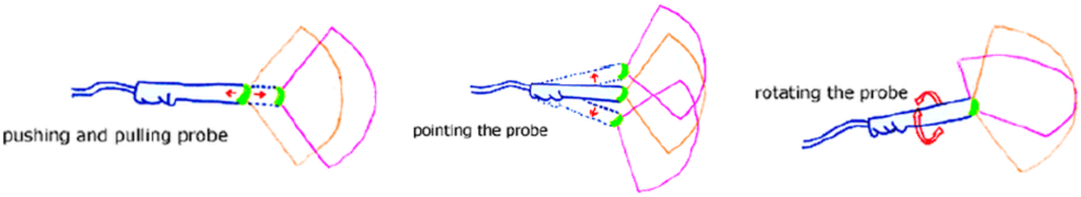


Figure 1.5: Desired direction

As ultrasound waves penetrate the body tissues of different acoustic impedance along the path of transmission, some are reflected back to the transducer (echo signals) and some continue to penetrate deeper. The echo signals returned from many sequential coplanar pulses are processed and combined to generate an image. Thus, an ultrasound transducer works both as a sound wave generator and as a sound wave receptor. The ultrasound pulse is short, but since it traverses in a straight path, it is often referred to as an ultrasound beam. The direction of ultrasound propagation along the beam line is called the axial direction, and the direction in the image plane perpendicular to axial is called the lateral direction. High -amplitude echoes will have high brightness presentation. In Figure 1.6, a B-Mode image ultrasound display can be seen and how a group of this images creates the 3D representation. The length of the scan lines determines the field of view (FOV). The distances will be in between 2 and 12 cm and so our images.

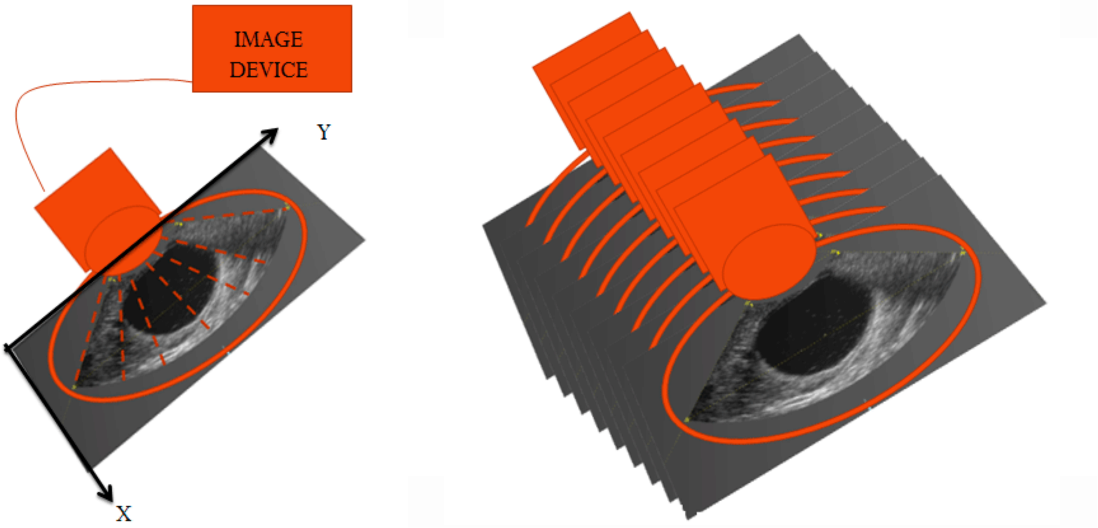


Figure 1.6: B Mode

1.3 Objective of the Thesis

The motivation of this research work begin through the university contact with a gynecologist doctor, who transmit us the real need of discriminating between benign and malignant adnexal masses in ultrasound images and his interest in image research. After finding information in several articles about the work that has been already done in this wide area, the opportunity of involving us with new ideas in this topic appeared.

Therefore, this project aims to develop a Computer Aided Diagnostic (CAD) technique of ultrasound images to discriminate between malignant and benign groups in order to be able to help as a practical tool during the diagnostic decision, with the future goal of assisting doctors in the interpretation of suspicious ultrasound images.

1.4 Structure of the Thesis

Chapter 2: introduces the image data sets and volumes.
Chapter 3: forms the first part of the thesis. The preprocessing algorithms for our set of images is described.
Chapter 4: presents the second part of the thesis. The extracted features calculated for each of the images are presented.
Chapter 5: indicates the dimensionality reduction process.
Chapter 6: forms the fourth part of the thesis. The classifier implementation is described and the final results are presented.
Chapter 7: forms the conclusion of the report and provides a short outlook.

This project has been performed in cooperation with the Clinical University of Navarre. The method has been modeled within MATLAB.

Chapter 2

Database

The image database has been taken from the gynecological department of the Clinical University of Navarre. 145 different volumes are used, each one obtained from a different patient. In Figure 2.1, we can see a helpful orientation of the corresponding volumes in order to differentiate between Front/Axial, Lateral and Top view. These orientations correspond to the three sections from the volume represented in Figure 2.2.

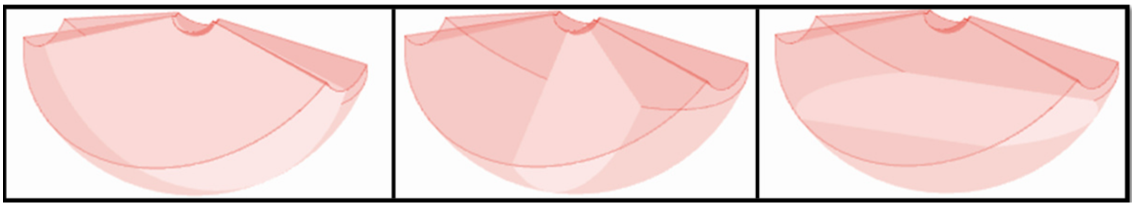


Figure 2.1: Front(A), Lateral(B) and Top view(C) from an Ultrasound Volume

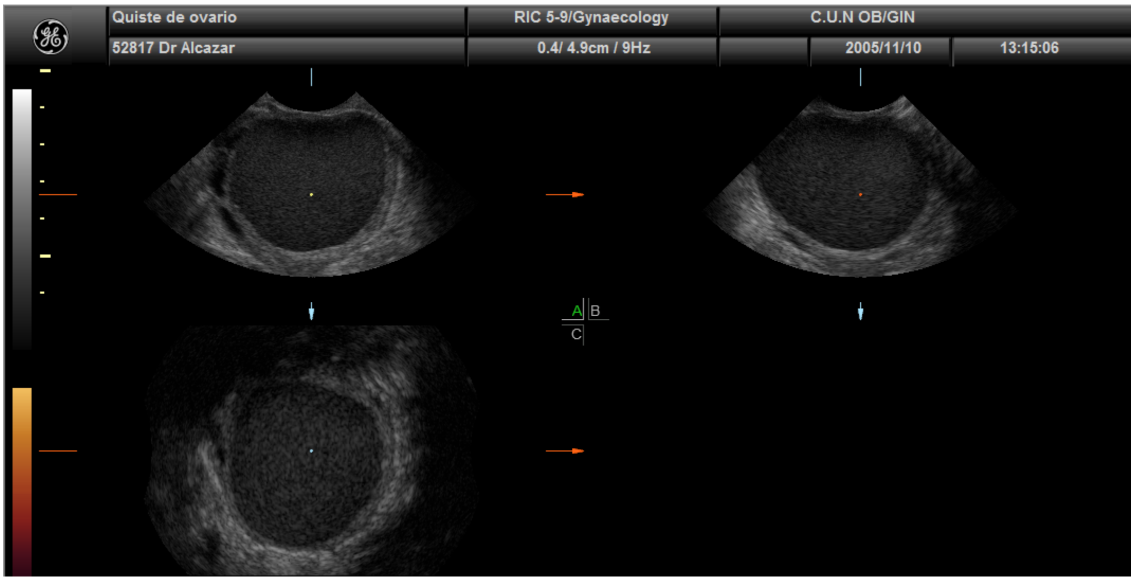


Figure 2.2: Ultrasound Volume

Apart from the patient and doctor identification, and the patient disease, there is also information about the sweep frequency of the ultrasonic system. As we have said before, the FOV is in the range between 2 and 12 cm.

The volumes have been taken from a Voluson ultrasound system (.V00 format). Therefore, the program that is going to be used to premanipulate this volumes will be $4DView$, a software used to optimize, manipulate and analyze volume ultrasound data offline [1]. With this program, some measures will be taken in order to be able of reading the volumes properly. The following and main investigation will be continued through MATLAB.

Together with the volumes, a "gold standard" value of every volume is given. It represents the diagnostic truth of each original image. In this way, from the 145 volumes, we can differentiate between 106 benign volumes and 39 malignant ones. The specific number of benign and malignant volumes has been chosen according to its probability of appearance in general population. Approximately, 75 to 85 % are benign adnexal masses. Therefore, this proportion will be used throughout this thesis.

The age of the women is also given by the gynecologist. The clinic experience reveals that women between 35 and 65 years old have more incidence of having an ovarian tumor and so are the ages taken into account for this project. Patients mean age is 43 years old.

2.1 Classification Types

Differentiation of adnexal masses into benign and malignant is based on many morphological parameters. Transvaginal ultrasound investigation of any adnexal mass provides information on its location in the pelvis, its laterality and its relation with the adjacent organs.

Among our database, we can differentiate the following classification:

Number	**Malignant Names**
23	Ovarian Tumor
4	Pelvic Tumor
1	Anexial Tumor
1	Ovarian Tumorization
1	Pelvic Tumorization
2	Ovarian Cysts
1	Ovarian Mass
1	Anexial Mass
1	Cervix neo
1	Ovarian Carcinoma
3	Without classification

Table 2.1: 39 Malignant Classification

Number	**Benign Names**
26	Ovarian Tumor
1	Pelvic Tumor
2	Anexial Tumor
1	Solid Tumor
14	Endometrioma
39	Ovarian Cysts
1	Hemorrhagic Cyst
1	Bilateral Ovarian Cysts
1	Anexial Cyst
3	Teratoma
1	Ovarian Tumor
1	Ovarian Lesion
1	Anexial Mass
14	Without classification

Table 2.2: 106 Benign Classification

As we can see, the more common are both Ovarian Tumor and Ovarian Cysts. Besides, they can be both benign or malignant. Pelvic Tumor is also frequent in both classifications. Transvaginal ultrasound investigation provides information on the morphology of the mass, classified into unilocular, multilocular, unilocular solid, multilocular solid, solid or unclassifiable. Information can also be obtained if a septum or multiple septa are present.

Chapter 3

Proposed System

From each of the 145 volumes, just one $2D$ image will be obtained in order to avoid redundancy. The goal is to simplify the problem by working with $2D$ images instead of with volumes. The volumes have been taken from a Voluson ultrasound system and they will be processed through MATLAB.

3.1 Preprocessing Method

First of all, the images that belong to each of the volumes will be read from Matlab. A set of cropped images will be obtained from the volume file and its correspondent volume mask. The images will be proportional to the original size. In Figure 3.1, it can seen how from the original volume, the cropped image and its mask are calculated.

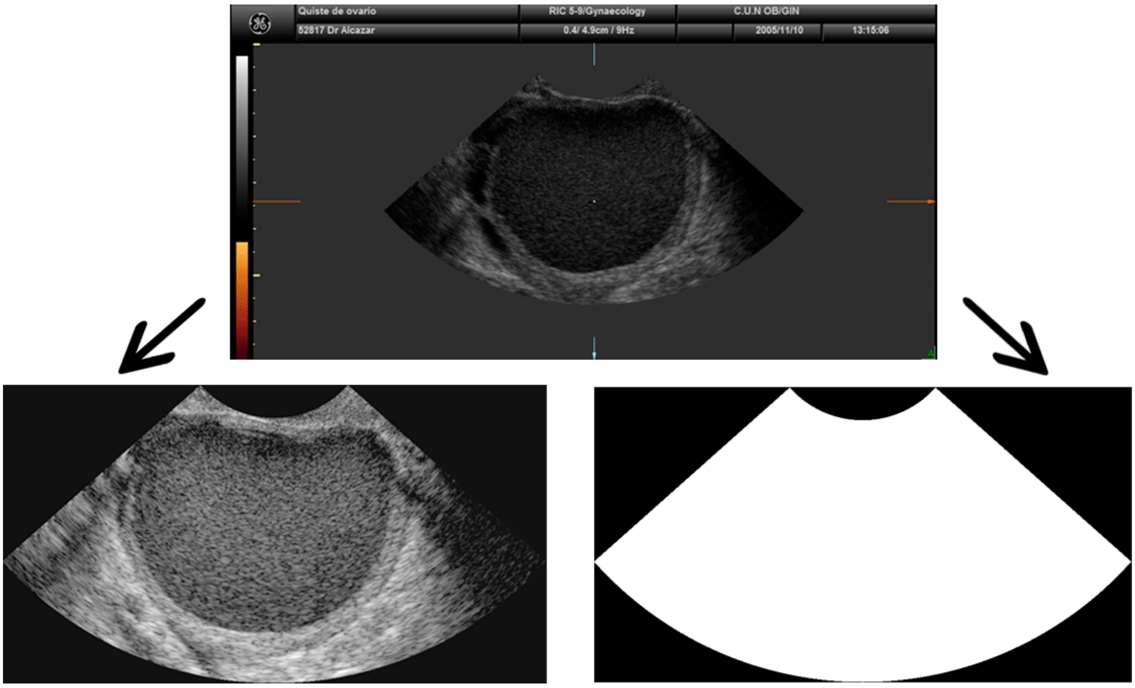


Figure 3.1: Image plane from the volume

The volumes have different sizes, depending on the adnexal mass. In general, it can be said that each volume has between 130 and 220 images. As we have said, one optimal image from each volume will be chosen, where the adnexal mass could perfectly be seen. A graphic example is presented in Figure 3.2.

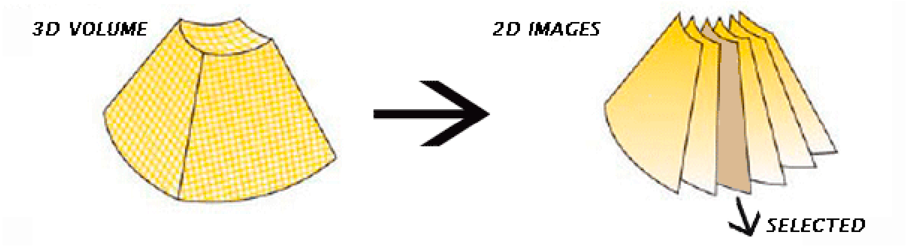


Figure 3.2: 3D to 2D Volumes

From now on, we will work with this optimal image from the whole volume. As detailed below, two different methods will be used, taking into account the whole image or just a defined region of interest (ROI). At the end, both results will be compared.

3.1.1 Semi-Automated Image Method

It is called semi-automated because the optimal image from the volume has been manually chosen. The image will be, first of all, multiplied by the mask in order to assure that no background information is used and later constrained to this area. The features from the following chapters will be applied to this region.

3.1.2 Non-Automated Method or ROI Method

As in Figure 3.3, the ROI of the optimal image of each volume is selected. This segmentation is supervised by the gynecologist expert in order to increase the reliability of the selection. The features from the following chapters will be also constrained to this area. Moreover, the image will be cropped by calculating the bounding box of the ROI mask. In this way, smaller images will be used in order to avoid computation time.

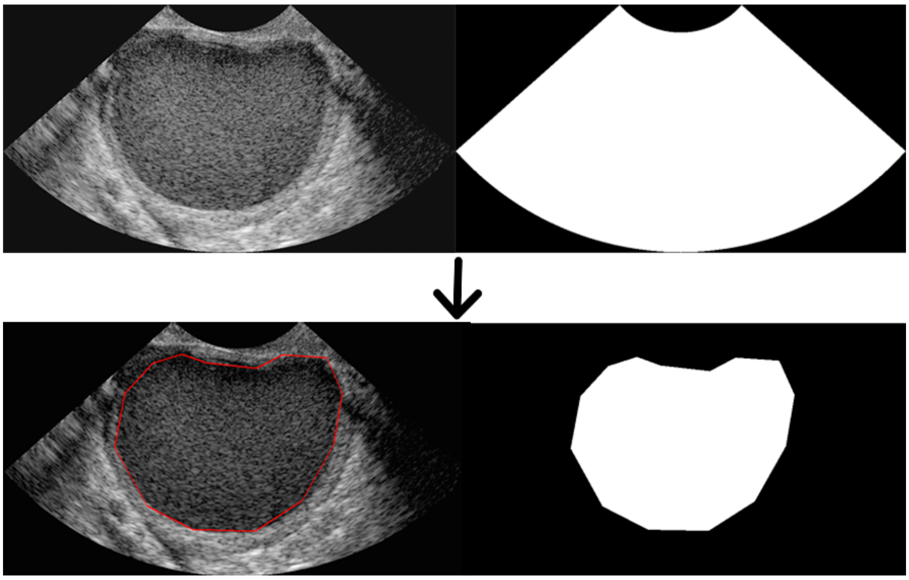


Figure 3.3: ROI Image plane from the volume

3.2 Diagram

The proposed system is indicated in Figure 3.4. Four main parts can be differentiated. In the first one, image extraction and preprocessing is made. Two groups are obtained: the Semi-automated, which is directly obtained, and the ROI ones, by selecting the specific region and verifying it with an expert.

The second part consists in a feature extraction process, where we obtain different texture features based on the bibliography, which will lead us to a posterior discrimination. The third part proposes a dimensionality reduction system, from where the significant features or a combination of them will be taken.

Finally, the neural network design is presented. As it will be later explained, the data is divided following the cross-validation method, in order to make a proper training and a posterior discrimination.

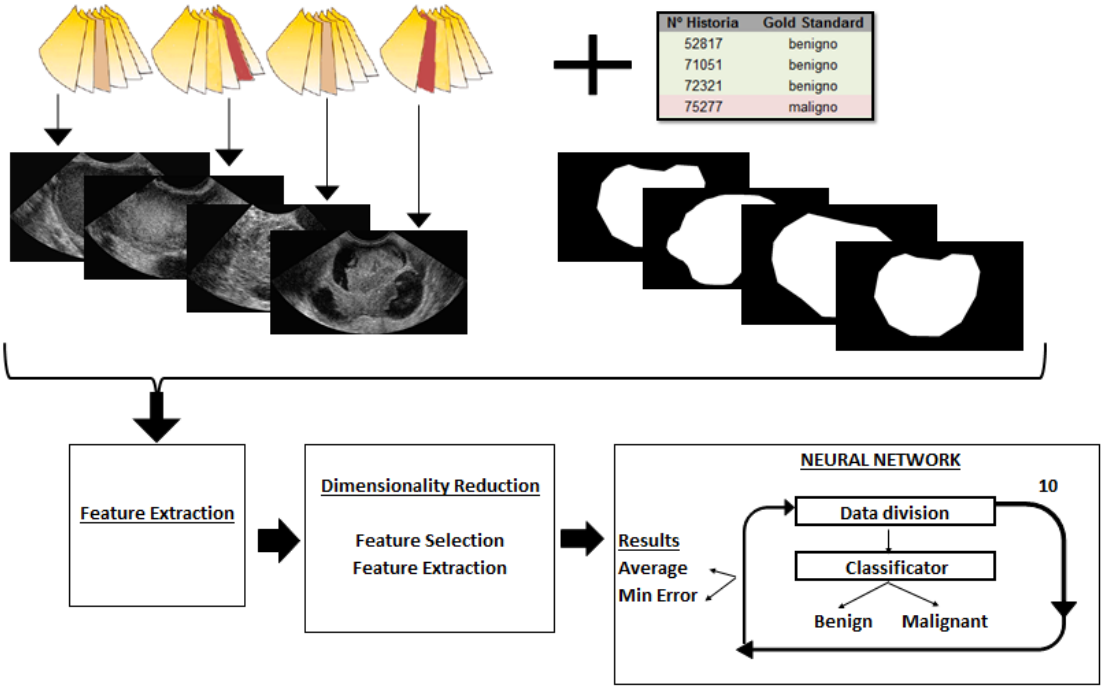


Figure 3.4: Block Diagram System

In Figure 3.5, it can also be seen how we are working with images of different size, which should be later taken into account. They have been rescaled to provide the same height and maintain their original proportions.

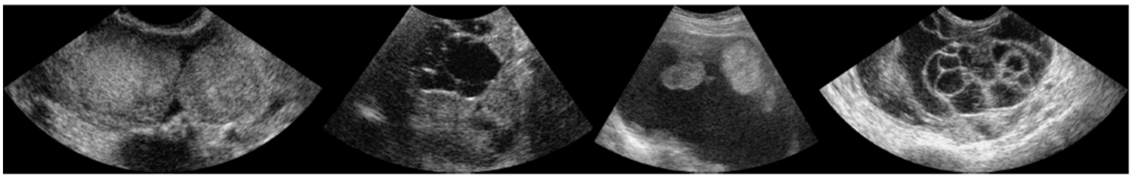


Figure 3.5: Equal image height

Chapter 4

Extracted Features

This chapter focus on the different operators that have been used in order to significantly characterize the image. This will give us important information of the image and it will lead us to a posterior classification. As we have said in the previous chapter, image files have been taken from a Voluson device software, which comes from the General Electric (GE) company. Therefore, although image proportions keep constant, original distances (mm/cm) are lost when reading the files in MATLAB. In other words, areas or perimeters from the different images cannot be compared so only texture operators will be calculated.

4.1 Local Binary Pattern (LBP)

The *Local Binary Pattern* is an efficient texture operator specially used to emphasize small scale image texture and highlight similarities between images. Theoretically, considering the gray level of an arbitrary pixel g_c of an image, the local binary pattern replaces each value of g_c with an 8-bit binary code. The input variables are the *center pixel* gray level g_c, the chosen *radius* R and the *number of neighboring pixels* P. Generally, a window of radius $1(R = 1 = 3x3)$ centered in g_c is chosen together with its $P = 8$ neighboring pixels in order to model small-scale image texture.

The method consists in calculating the difference between the center pixel g_c and its P neighboring pixels $(g_p, p = 0, .., P - 1)$. A clockwise direction is followed, starting up with the pixel on the right side of g_c. Then a thresholding function $s(x)$ is applied. If the difference value is lower than 0, 0 is assigned. Otherwise, 1. Finally, the corresponding LBP image is obtained by translating the binary code into its decimal value. In Figure 4.1, we can see a practical example. This feature can be expressed mathematically by:

$$LBP_{P,R} = \sum_{p=0}^{P-1} s(g_p - g_c)2^p \qquad s(x) = \begin{cases} 1 & \text{si } x >= 0 \\ 0 & \text{otherwise} \end{cases} \tag{4.1}$$

4.1.1 Histogram calculation

The first option will be to calculate the normalized histogram of the LBP image. Considering a gray level histogram $h_i, i = 0, 1, ..., L - 1$, where L is the number of distinct gray levels, the normalized histogram will be $H_i, i = 0, 1, ..., L - 1$, where $H_i = h_i/MN$ and MN are the dimensions of the image. Two different scales will be used: (P=8, R=1) and (P=8, R=2). Their design is shown in Figure 4.2. The histogram bins can be directly used as a significant feature, as applied in the paper of Khazendar, et al [4]. The histogram difference between a real image and a LBP image can be seen in Figure 4.4. Indeed, small scale information is accented. Besides, the skewness and the variance parameters of the LBP histogram image will also be calculated as features.

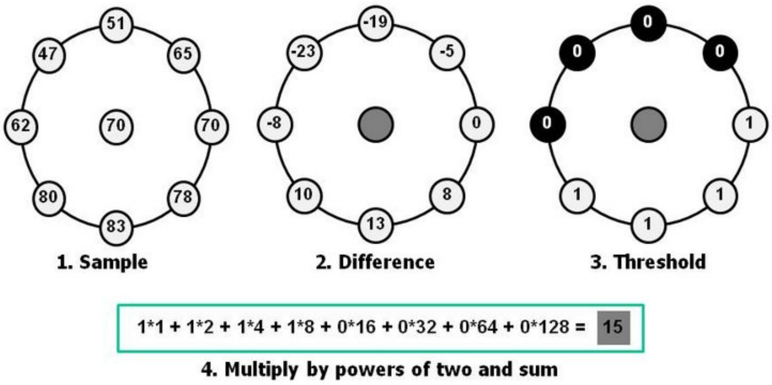


Figure 4.1: LBP Example

On one hand, the skewness parameter is a measure of the asymmetry of the data around the sample mean. If skewness is negative, the data are spread out more to the left of the mean than to the right. If skewness is positive, the data are spread out more to the right. The skewness of the normal distribution (or any perfectly symmetric distribution) is zero. On the other hand, the variance measures how far a set of numbers is spread out. A variance of zero indicates that all the values are identical. Variance is always non-negative: a small variance indicates that the data points tend to be very close to the mean (expected value) and hence to each other, while a high variance indicates that the data points are very spread out around the mean and from each other.

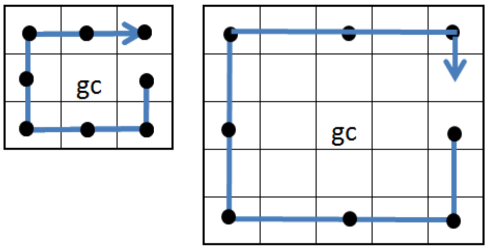


Figure 4.2: LBP Scales: 81, 82

In Figure 4.4, the zero mean histograms (LBP 81) from a benign and malignant ROI image are represented. It can clearly be seen how the malignant images (in red) seem to be more compact than the benign ones (in blue). Therefore, the variance parameter could probably lead to significant results. However, its significance will be measured in the following chapter.

4.1.2 Average Power and Entropy calculation

Multi-scale analysis of the image using LBP can also be done. The LBP images will be calculated using three different scales: $P = 8, R = 1(3x3); P = 16, R = 2(5x5); P = 32, R = 3(7x7)$, as applied in [10]. The scales design can be seen in Figure 4.5. Average power and entropy of the resulting LBP image will be used as features.

The average power, is related to the mean square value of the image normalized by the number of samples in the image. In other words, it can be said that it is related with its intensity. The more intensity in the pixels of the image, the more average power the image will have, which can clearly be seen in the example of Figure 4.6. Entropy will be in the following sections explained, as an independent parameter.

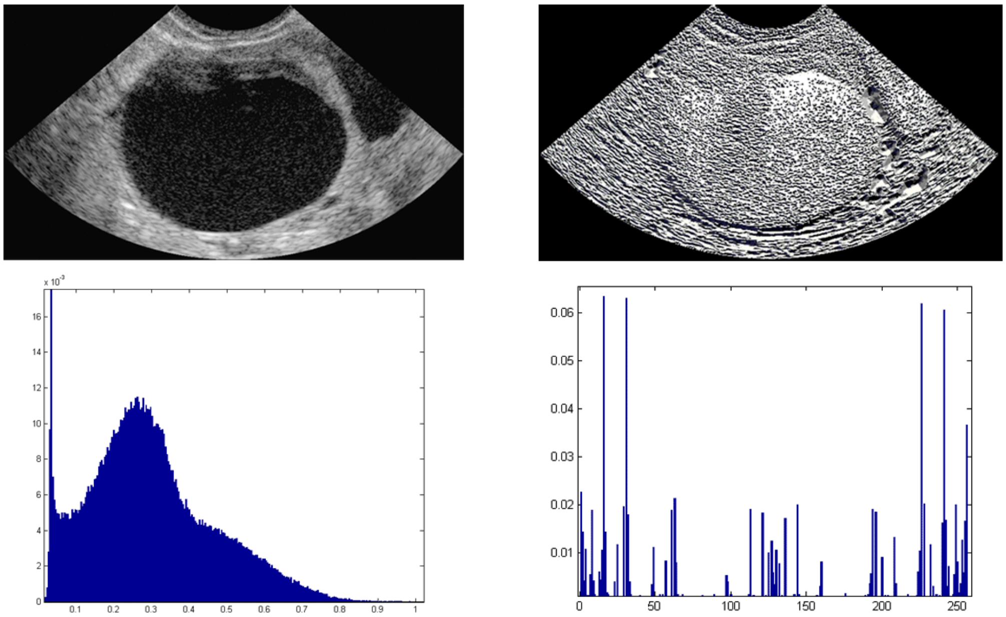


Figure 4.3: LBP Histogram 81 Example

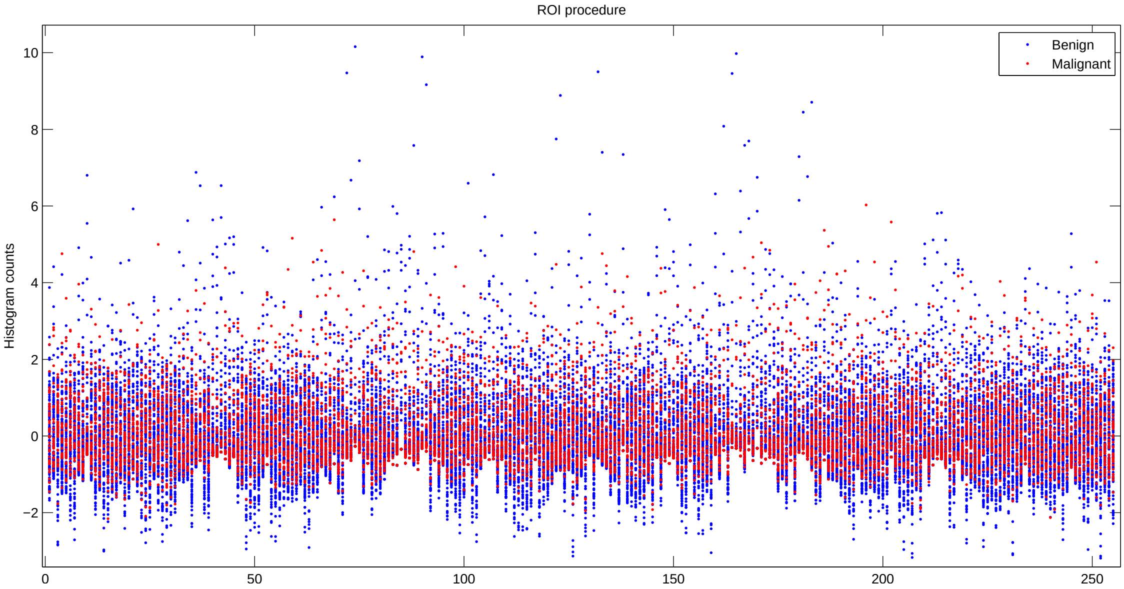


Figure 4.4: LBP Histogram 81. Benign and Malignant. ROI

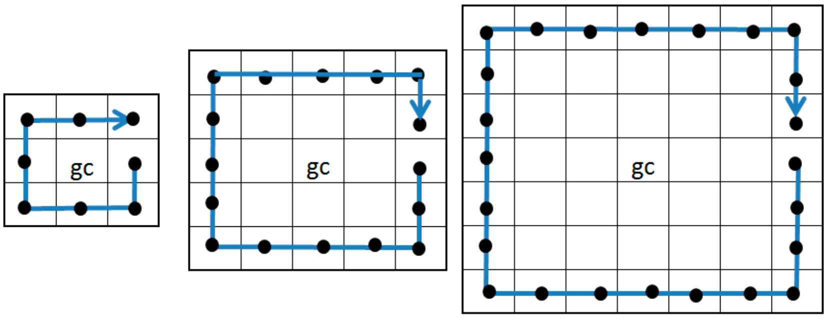
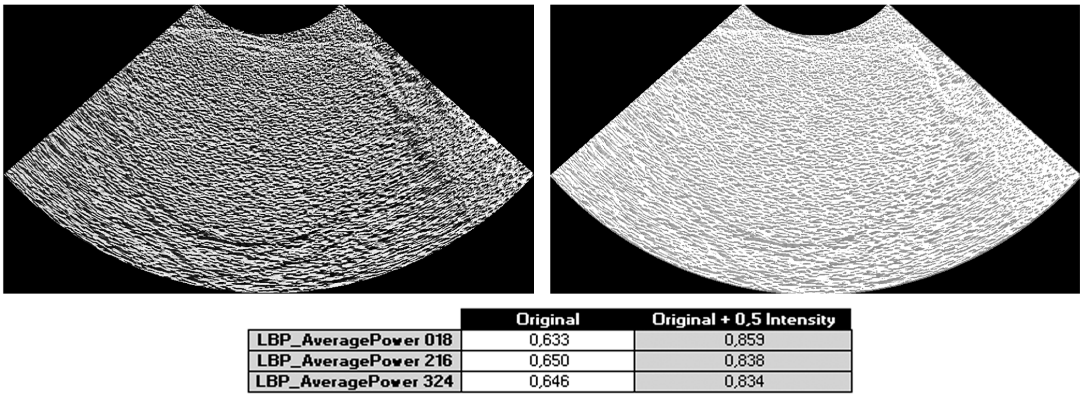


Figure 4.5: LBP Scales: 81, 162, 323



	Original	Original + 0,5 Intensity
LBP_AveragePower 018	0,633	0,859
LBP_AveragePower 216	0,650	0,838
LBP_AveragePower 324	0,646	0,834

Figure 4.6: LBP Average Power Example

4.2 Fractal Dimension (FD)

The *fractal dimension* (FD) is a real number used to characterize the geometric complexity of a fractal. The simplest example of a fractal structure will be a human body. A human body consists of organs; an organ consists of large leaves; a large leaf consists of small leaves; a small leave consists of cell units and so on. All these structures are different sized units with self-resemblance, and that is what defines a fractal structure. Therefore, a fractal structure can be classified into different levels by using its sized units. In Figure 4.7, we can see a fractal surface divided into its different levels from the lowest to the highest one, where r is the size of the structure and $N(r)$ is the total number of units needed to cover a certain structure.

Defining a *fractal* as a bounded set A for which the fractal dimension (morphologic complexity) is strictly larger than the topological dimension (D), a scale factor s ($s = 1/r$), and a number of pieces N, the following expression can be obtained. A would be self-similar if it is the union of N(r) non-overlapping copies of itself scaled up or down by a factor of r.

$$FD = \frac{log(N(r))}{log(\frac{1}{r})} \tag{4.2}$$

As in [8], this feature is used to express self-similarity and give information about the irregularity of the pixels of an image. One way to quantify FD is the **Box-counting** method. It consists in covering the image with a non-overlapping grid made of boxes and then counting how many boxes of the grid are covering our image. After that, this method is repeated by iterating the process using a finer grid. At the end, the pattern of how $N(r)$ changes with r will be obtained, as represented in the following graph. Using a logarithmic scale, a lineal regression model will be used to fit the line. The FD value is given by the slope of this line. The input can be a $1D$ segment, a $2D$ image or a $3D$ volume. Certainly, smaller squares will pick up more detail, and will give a better approximation of the shape ($N(r)$ squares of side length r).

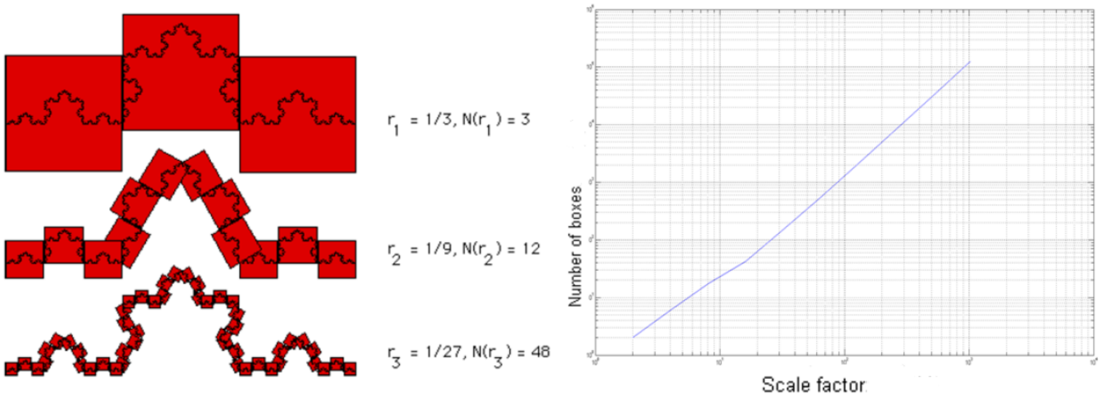

r_1 = 1/3, N(r_1) = 3
r_2 = 1/9, N(r_2) = 12
r_3 = 1/27, N(r_3) = 48
Number of boxes
Scale factor

Figure 4.7: FD Example (Scale factor ($1/r$) and Number of boxes ($N(r)$))

In this sense, the more irregular the surface is, the higher the value will be. In Figure 4.8, an example can be seen. Small squares are used on the left image, while bigger ones are used on the right. This example show how the box grid is applied to the non-zero elements of the image and how smaller boxes provide more detail than the bigger ones. However, in our case, covering the image surface so as to count the number of boxes gives no extra information about the irregularity of the pixels of the adnexal mass itself.

Therefore, as in [5], the **Modified differential box counting method** will be used in order

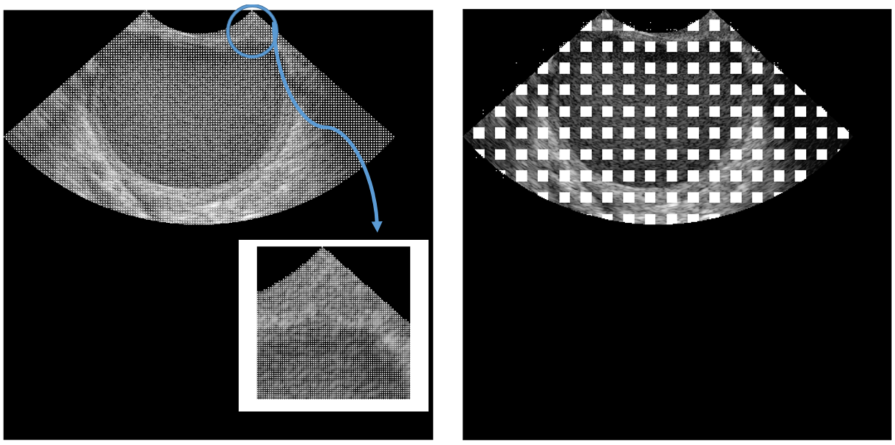


Figure 4.8: Box Counting Method Example

to get information about the pixel intensities itself. First of all, for efficient computation, zero padding will be applied to our image in order to work with a power of 2 grid size. Then, considering an image $I(i,j)$ of size MxM $(i = 1,..M; j = 1,..M)$, it will be scaled down to a certain number of rxr grids, whose side length r is in between 2 and $M/2$. As in Figure 4.9, each grid can be viewed as a column of boxes of size $rxrxr'$ placed one above the other, indicating different gray levels values.

Considering G as the total number of gray levels (256 in this case), the maximum (gk) and minimum (gl) gray levels will be taken for each (i,j)th grid. This values will be saved in the kth and lth boxes respectively and n_r will in this case not be the number of non-overlapping copies but the count of the number of boxes on the top of the (i, j)th block, $n_r(i,j) = gk - gl + 1$. In this way, the image is seen as a 3-D landscape and the total contribution of the volume is equivalent to N_r boxes, where:

$$N_r = \sum_{i,j} n_r(i,j) \tag{4.3}$$

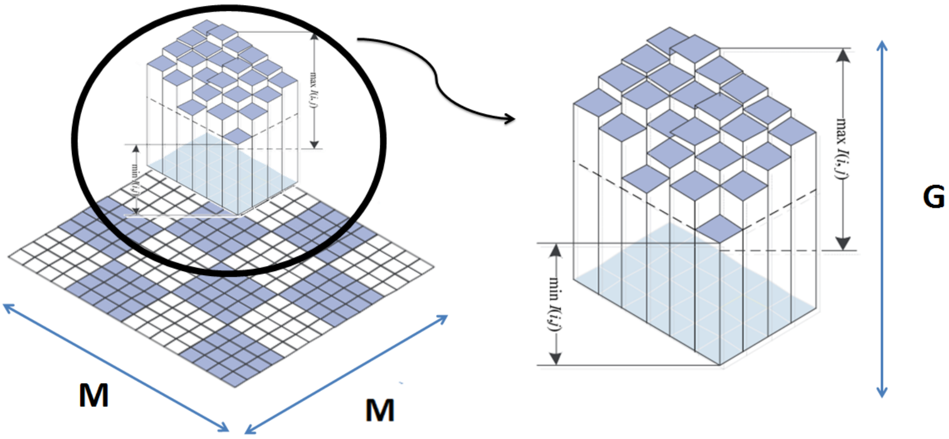


Figure 4.9: Differential Box Counting Method Example, [13]

From Equation 4.2, using the least square error linear fit for $logN_r$ against $log(1/r)$,the fractal dimension can be obtained as the slope of the fitted line. In Figure 4.10, three iterations can be seen, which have $2^2, 2^3$ and 2^4 boxes respectively. Above all these boxes, we will also have a

$(i, j)th$ grid, from where the intensities difference of each box will be taken.

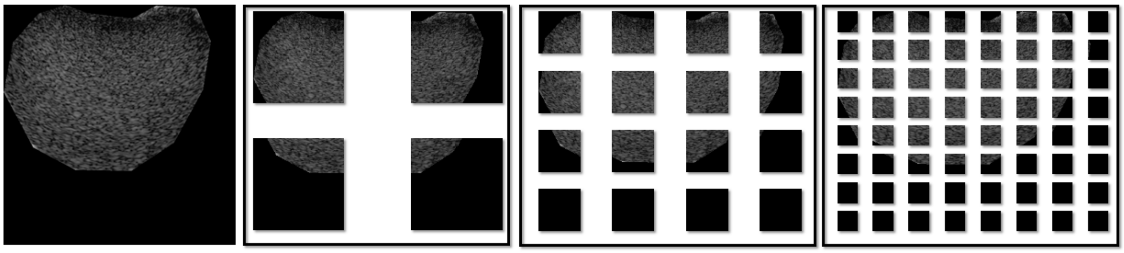
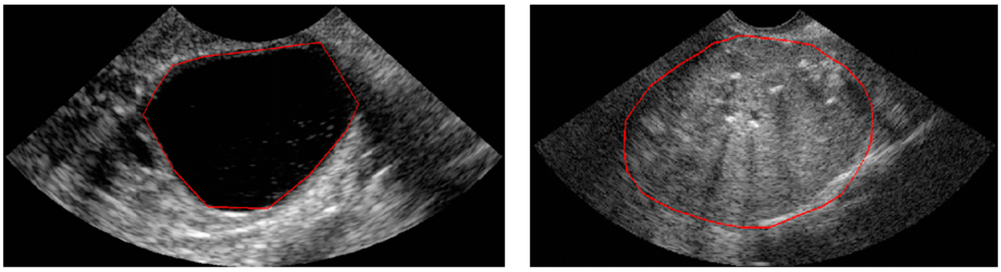


Figure 4.10: Box Counting Method ROI Example

A scale-independent verification of this feature can be made. Taking into account the image of the example above and its half sized version, the FD value maintains invariant (2,500 and 2,499). Therefore, it can be explained that the texture keeps equal in both images, despite its image size.

In Figure 4.11, two different ROI segmentations with a red line can be seen. One of them, tends to have a smoothies gray level surface, while the other has a more irregular one. Therefore, the FD values are different: the value of the image on the right is higher than the one of the image on the left. Both are normalized.



	SMOOTH SURFACE	IRREGULAR SURFACE
FD	0,06	0,87

Figure 4.11: FD Example

4.3 Entropy (E)

Entropy is a statistical measure of randomness that can be used to characterize the texture of an input image, [7]. The image histogram carries important information about the content of an image and can be used for discriminating both groups. Considering the gray level histogram $h_i, i = 0, 1, ..., L-1$, L is the number of distinct gray levels. If MN are the dimensions of the image, the normalized histogram is $H_i, i = 0, 1, ..., L-1$, where $H_i = h_i/MN$. Theoretically, Shannon Entropy can be defined as:

$$E = -\sum_{i=0}^{L-1} (H_i \log_2(H_i)) \tag{4.4}$$

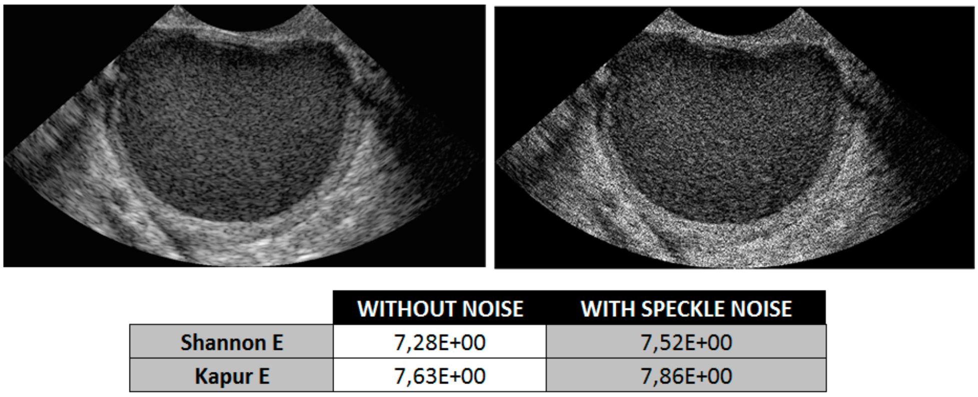
where H_i contains the probabilities from each level of the histogram calculation. In most feature descriptors, Shannon's measure is used to measure entropy. However, in this project non-Shannon measures are also used because they have a higher dynamic range over scattering conditions.

Therefore, they are useful in estimating scatter density and regularity. Kapur's measure can be defined as:

$$EK = \frac{1}{\beta - \alpha} \log_2 \left(\frac{\sum_{i=0}^{L-1} H_i^{\alpha}}{\sum_{i=0}^{L-1} H_i^{\beta}} \right) \tag{4.5}$$

In this project, as mentioned in [7], we consider $\alpha = 0.5$ and $\beta = 0.7$.

Therefore, flat images or those with a uniform distribution of gray levels have a low entropy. However, images with random noise have more entropy due to the fact that they are not highly spatially correlated or, in other words, complex to compress. In Figure 4.12, an example between the original image (left) and the same image with Speckle noise can be seen. Speckle is a granular noise that inherently exists and degrades the quality of medical ultrasound images. As we expected, the entropy values differ and the image with noise presents a higher entropy value.



	WITHOUT NOISE	WITH SPECKLE NOISE
Shannon E	7,28E+00	7,52E+00
Kapur E	7,63E+00	7,86E+00

Figure 4.12: E Example

4.4 Hu's Invariant Moments (IM)

An *image moment* is a certain particular weighted average of the image pixels intensities, or a function of such moments, usually chosen to achieve a good interpretation or property. Therefore, image moments are very useful to describe objects after segmentation, [7]. It is possible to obtain properties such as area or intensity, centroid and orientation information. The Hu's Seven Invariant Moments are invariant under translation, scaling, and also rotation. Therefore, they describe the image despite of its location, size, and orientation.

Theoretically, for a $2D$ digital image $f(x, y)$ the central moment order $(p + q)$ is defined as:

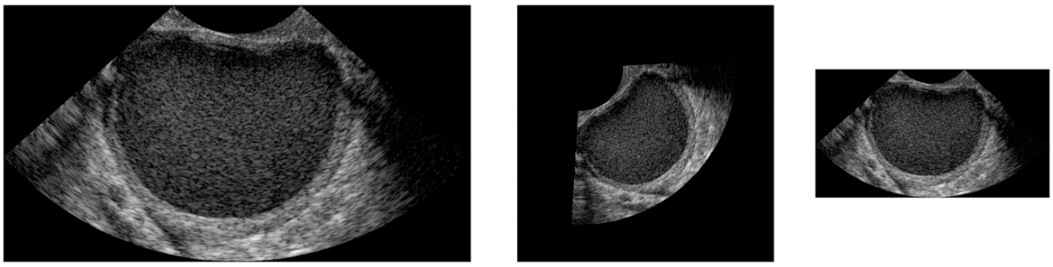
$$\mu_{pq} = \sum_{x} \sum_{y} (x - \hat{x})^p (y - \hat{y})^q f(x, y) \tag{4.6}$$

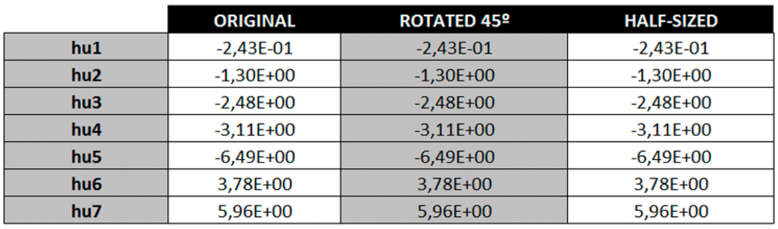
where $\hat{x} = \frac{m_{10}}{m_{00}}$ and $\hat{y} = \frac{m_{01}}{m_{00}}$ are the centroid of the image. Therefore, central moments are independent from their position. In the same way, the normalized central moment of order $(p+q)$ is defined as: $\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^{\gamma}}$ where $\gamma = \frac{p+q}{2}$. From this normalized central moment, Hu defined seven values up to three $(p, q : 0, 1, 2, 3)$ that are invariant. They are calculated as in the following equations.

$$hu_1 = (\eta_{2,0} + \eta_{0,2})$$

$$\begin{aligned}
hu_2 &= (\eta_{2,0}-\eta_{0,2})^2+4\eta_{1,1}^2 \\
hu_3 &= (\eta_{3,0}-3\eta_{1,2})^2+(3\eta_{2,1}-\eta_{0,3})^2 \\
hu_4 &= (\eta_{3,0}+\eta_{1,2})^2+(\eta_{0,3}+\eta_{2,1})^2 \\
hu_5 &= (\eta_{3,0}+3\eta_{1,2})(\eta_{3,0}+\eta_{1,2})[(\eta_{3,0}+\eta_{1,2})^2-3(\eta_{0,3}+\eta_{2,1})^2]+ \\
&\quad (3\eta_{2,1}+3\eta_{0,3})(\eta_{0,3}+\eta_{2,1})[3(\eta_{3,0}+\eta_{1,2})^2-(\eta_{0,3}+\eta_{2,1})] \\
hu_6 &= (\eta_{2,0}-\eta_{0,2})[(\eta_{3,0}+\eta_{1,2})^2-(\eta_{0,3}+\eta_{2,1})^2]+ \\
&\quad 4\eta_{1,1}(\eta_{3,0}+\eta_{1,2})(\eta_{0,3}+\eta_{2,1}) \\
hu_7 &= (3\eta_{2,1}-\eta_{0,3})(\eta_{3,0}+\eta_{1,2})[(\eta_{3,0}+\eta_{1,2})^2-3(\eta_{0,3}+\eta_{2,1})^2)]- \\
&\quad (\eta_{3,0}-3\eta_{1,2})(\eta_{0,3}+\eta_{2,1})[(3\eta_{3,0}+\eta_{1,2})^2-(\eta_{0,3}+\eta_{2,1})^2]
\end{aligned} \tag{4.7}$$

In Figure 4.13, an example can be seen. The original image, its 45 grades rotated image version and its half sized image version, together with the seven Hu moments. The $sign(hu)\cdot log_{10}(hu)$ of the absolute result has been taken in order to reduce the dynamic range. As it can be seen, the values keep invariant.





	ORIGINAL	ROTATED 45º	HALF-SIZED
hu1	-2,43E-01	-2,43E-01	-2,43E-01
hu2	-1,30E+00	-1,30E+00	-1,30E+00
hu3	-2,48E+00	-2,48E+00	-2,48E+00
hu4	-3,11E+00	-3,11E+00	-3,11E+00
hu5	-6,49E+00	-6,49E+00	-6,49E+00
hu6	3,78E+00	3,78E+00	3,78E+00
hu7	5,96E+00	5,96E+00	5,96E+00

Figure 4.13: IM Example

4.5 Gray Level Co-ocurrence Matrix (GLCM)

Another texture measure, that will give us information about the spatial arrangement of the intensities of our image, is the *Gray Level Co-ocurrence Matrix*.

The elements of GLCM, $C_d(i,j)$, are made up of the relative number of times the gray level pair (i,j) occurs when pixels are separated by the distance $(1,0)$, which corresponds to the vertical direction as we can see in Figure 4.14 together with an example. The index i will go over the rows of the image, and j will go over the columns of the image. This probability is measured by the following equation, which corresponds to the normalization of the GLCM, making the total sum equal to one.

$$P_d(i,j)=\frac{C_d(i,j)}{\sum_i\sum_j C_d(i,j)} \tag{4.8}$$

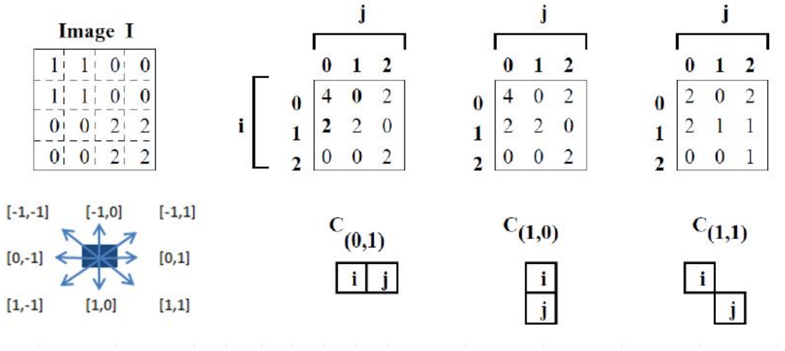
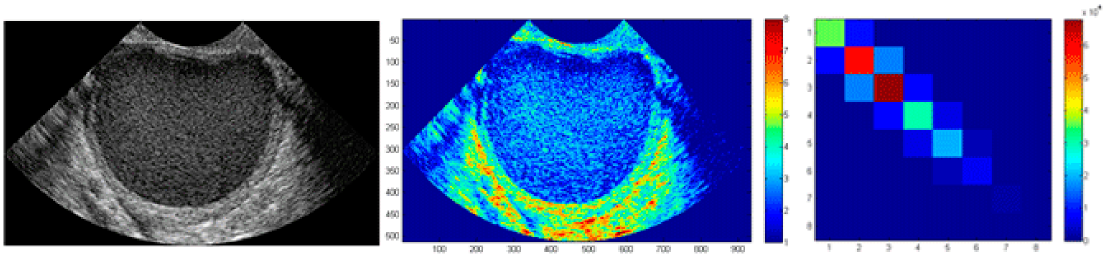


Figure 4.14: 3 GLCM examples for a gray-tone image

In Figure 4.15, the calculation process can be seen. First of all, the original image will be scaled to 8 gray levels. The scaled image is represented in pseudocolor by mapping each intensity value to a certain color map, in order to better see the scaled version of the gray scale image. After that, the GLCM will be calculated. The matrix dimensions are 8x8.



```
0.1163    0.0282    0.0001         0         0         0         0         0
0.0279    0.1918    0.0546    0.0002         0         0         0         0
0.0001    0.0540    0.2243    0.0269    0.0002         0         0         0
     0    0.0001    0.0261    0.0995    0.0183    0.0001         0         0
     0         0    0.0001    0.0178    0.0634    0.0090    0.0000         0
     0         0         0    0.0001    0.0088    0.0237    0.0019    0.0000
     0         0         0         0    0.0000    0.0019    0.0036    0.0002
     0         0         0         0         0    0.0000    0.0002    0.0004
```

Figure 4.15: GLCM calculation process: Original, Pseudocolor and GLCM

However, although the co-occurrence matrix captures properties of a texture, it is not directly useful for further analysis, such as comparing two textures. Therefore, numeric features such as correlation [8], entropy and Moment 4 [9] are computed instead. From the normalized GLCM or also called probability matrix ($P_{i,j}$), the following equations are obtained.

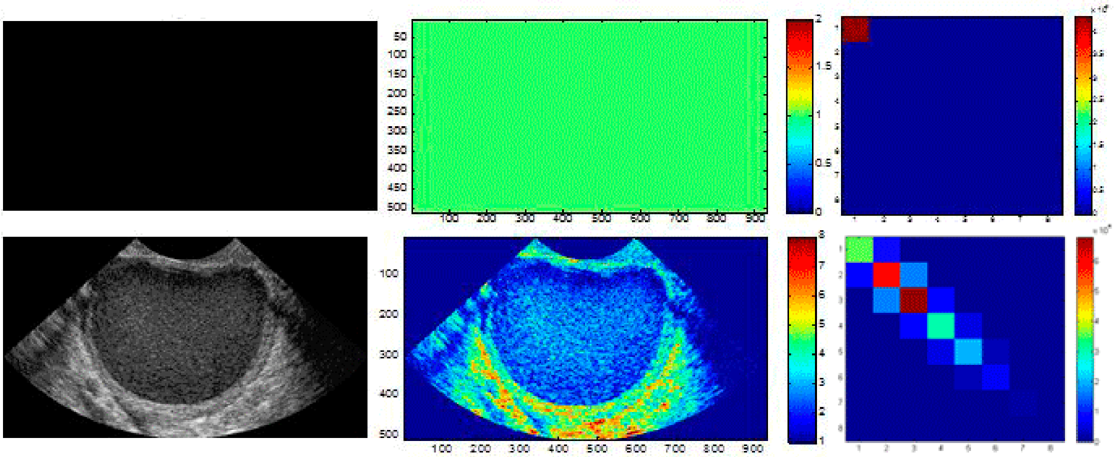
$$\text{GLCM Correlation} = \sum P_{i,j} \frac{(i-\mu_i)(j-\mu_j)}{\sqrt{(\sigma_i^2)(\sigma_j^2)}} \tag{4.9}$$

$$\text{GLCM Entropy} = -\sum_i \sum_j P_{i,j} log(P_{i,j}) \tag{4.10}$$

$$\text{GLCM M4} = \sum_i \sum_j (i-j)^4 P_{i,j} \tag{4.11}$$

where μ_i and μ_j are the mean and σ_i and σ_j the variance of the gray level appearance i and j, respectively.

In Figure 4.16, we can see an example where the entire process is presented. We have the original images (a uniform and a original one), the scaled images with the pseudocolor mapping and its GLCM, from where we will calculate the explained features.



	UNIFORM	NOT UNIFORM
GLCM Correlation	NaN	9,48E-01
GLCM Entropy	0,00E+00	2,00E+00
GLCM M4	0,00E+00	1,23E+00

Figure 4.16: GLCM Correlation Example

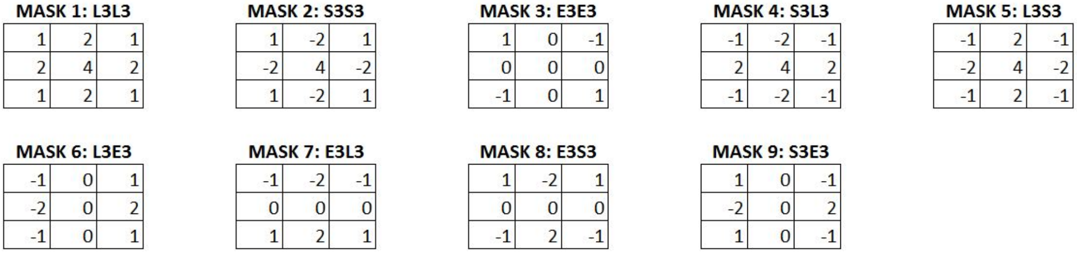
In the correlation measure, it can be seen that 0 is uncorrelated and 1 is perfectly correlated. When an image area is completely uniform, the GLCM variance is 0, just as the first-order image variance. As a result, the denominator of the correlation equation becomes 0, and correlation becomes undefined (NaN: Not a Number). The entropy measure is also related to the grade of variability in the image. The uniform image has only one non-zero pixel in the GLCM. Therefore the entropy and M4 are zero. The Moment 4 presents also a zero value in the uniform image.

4.6 Laws Texture Energy (LTE)

Laws Texture Energy is another approach to generating texture features. In this case, local masks will be used to detect various types of texture and then estimating its energy. As in paper [10], a set of three 1D vector convolution masks is used to compute it. Their names, $(L3, E3, S3)$, will describe its function, the level, edge and spot feature respectively.

$$L3(Level) = [1,2,1] \;\; E3(Edge) = [-1,0,1] \;\; S3(Spot) = [-1,2,-1] \tag{4.12}$$

As we can see in Figure 4.17, by the convolution of these 1D horizontal vectors with vertical ones, nine $2D$ masks (3x3) will be obtained. All the possible combinations are: L3L3 (mask1), S3S3 (mask2), E3E3 (mask3), S3L3 (mask4), L3S3 (mask5), L3E3 (mask6), E3L3 (mask7), E3S3 (mask8) and S3E3 (mask9). Due to the fact, that all these masks have a zero mean, except from L3L3, only eight masks will be used and the mask 1 (L3L3) will normalized the contrast of all other texture images. There are certain symmetric masks, for example, masks 4 and 5, masks 6 and 7, masks 8 and 9. Each of them measure horizontal and vertical content respectively. Anyway, we will work with them separately.



MASK 1: L3L3

1	2	1
2	4	2
1	2	1

MASK 2: S3S3

1	-2	1
-2	4	-2
1	-2	1

MASK 3: E3E3

1	0	-1
0	0	0
-1	0	1

MASK 4: S3L3

-1	-2	-1
2	4	2
-1	-2	-1

MASK 5: L3S3

-1	2	-1
-2	4	-2
-1	2	-1

MASK 6: L3E3

-1	0	1
-2	0	2
-1	0	1

MASK 7: E3L3

-1	-2	-1
0	0	0
1	2	1

MASK 8: E3S3

1	-2	1
0	0	0
-1	2	-1

MASK 9: S3E3

1	0	-1
-2	0	2
1	0	-1

Figure 4.17: LTE Masks

Texture is a set of primitive texels (texture pixel, fundamental unit of texture space) in some regular or repeated relationship. To obtain the texture image (TI), we will convolve the original image with all those 2D local masks in order to detect the different types of texture. Therefore, the result is a full image, representing the application of the *kth* mask to the input one. For example, the texture image and its normalization for mask 3 (E3E3) will be:

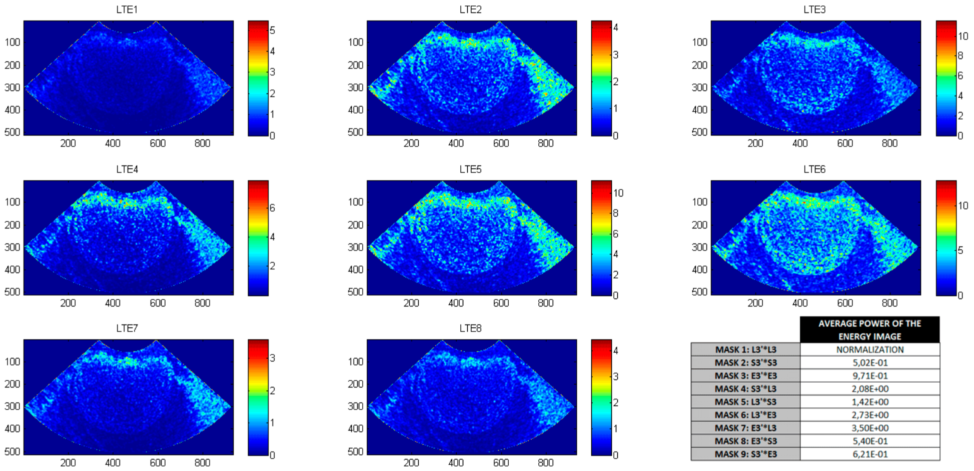
$$TI_{\text{E3E3}} = I * \text{E3E3} \tag{4.13}$$

$$TI_{\text{E3E3 Normalized}} = \frac{TI_{\text{E3E3}}}{TI_{\text{L3L3}}} \tag{4.14}$$

where * denotes the convolution operator. The resultant normalized TI images are passed through an average filter of absolute values. The texture energy (TE) map is obtained:

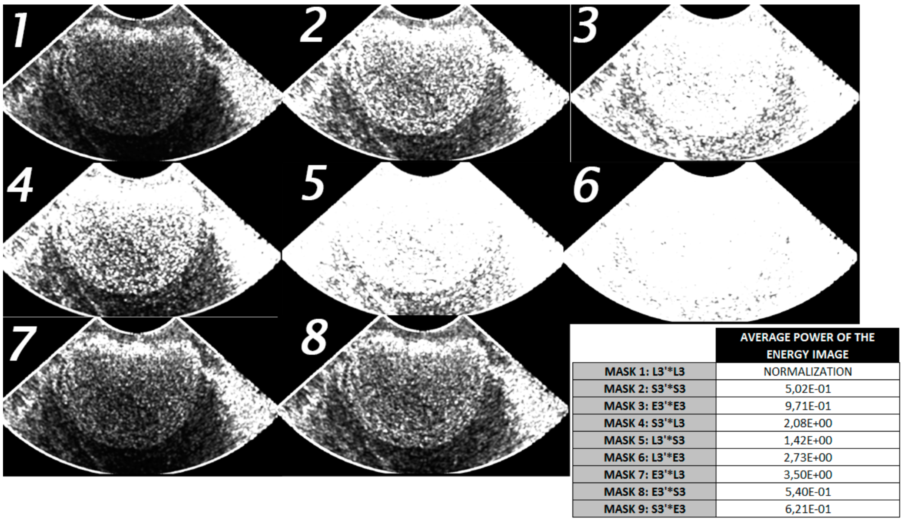
$$TE(i,j) = \sum_{-3}^{3}\sum_{-3}^{3} |TI_{i+u,j+v}| \tag{4.15}$$

where u,v are the image dimensions. In Figure 4.18 and 4.19, the results from the eight texture image masks in a pseudo-color image and in gray scale can be seen. As feature, the **average power** will be calculated from them.



	AVERAGE POWER OF THE ENERGY IMAGE
MASK 1: L3'*L3	NORMALIZATION
MASK 2: S3'*S3	5,02E-01
MASK 3: E3'*E3	9,71E-01
MASK 4: S3'*L3	2,08E+00
MASK 5: L3'*S3	1,42E+00
MASK 6: L3'*E3	2,73E+00
MASK 7: E3'*L3	3,50E+00
MASK 8: E3'*S3	5,40E-01
MASK 9: S3'*E3	6,21E-01

Figure 4.18: LTE Example



	AVERAGE POWER OF THE ENERGY IMAGE
MASK 1: L3'*L3	NORMALIZATION
MASK 2: S3'*S3	5,02E-01
MASK 3: E3'*E3	9,71E-01
MASK 4: S3'*L3	2,08E+00
MASK 5: L3'*S3	1,42E+00
MASK 6: L3'*E3	2,73E+00
MASK 7: E3'*L3	3,50E+00
MASK 8: E3'*S3	5,40E-01
MASK 9: S3'*E3	6,21E-01

Figure 4.19: LTE Example

4.7 Gabor Wavelet Transform (GW)

Gabor filters through the Gabor Wavelet Transform, provide both frequency (stationary and periodic structure) and spatial locality. This is achieved by the convolution of our image with Gabor filters of different scales and orientations. Therefore, they allow the analysis of spatial variation in a similar way to the human visual system. For all these reasons, it has been proposed as a texture discrimination model.

In the spatial domain, a 2D Gabor filter is a Gaussian kernel or envelope modulated by a sinusoidal wave along the x-axis. This filter has both a real and an imaginary component. We can work with the complex number or use the real or imaginary part individually. As in [11], the Gabor filter gf can be defined by:

$$gf_{(l,k)}(m,n) = \frac{f^2}{\pi\gamma\eta}\exp(-x'^2\frac{f^2}{\gamma^2} + y'^2\frac{f^2}{\eta^2})\exp(j2\pi f x') \tag{4.16}$$

where f is the frequency of the sinusoidal factor, $x' = m\cos(\theta) + n\sin(\theta)$ and $y' = -m\sin(\theta) + n\cos(\theta)$. And where θ represents the orientation of the normal to the parallel stripes of a Gabor function, γ is the sharpness along the major axis X and η is the sharpness along the minor axis Y. l and k are the integers that identify the scale and orientation factor respectively and (m,n) are the image dimensions. The aspect ratio of the Gaussian is $\lambda = \eta/\gamma$.

Typically, a bank of Gabor filters is used. They are Gaussians of different sizes modulated by sinusoidal plane waves of different orientations. The kernel size is proportional to the width of the Gaussian. At least, it should be six times the standard deviation and the closest odd number should be taken. The Gabor wavelet representation $x_{l,k}$ of an image is the convolution of the image with a family of Gabor wavelets. In our work, we will use a set of 24 complex Gabor wavelets: 4 Scales ($l = 1:4$) and 6 Orientations ($k = 1:6$), in particular: $0°, 30°, 60°, 90°, 120°$ and $150°$. The Gabor results can be defined as:

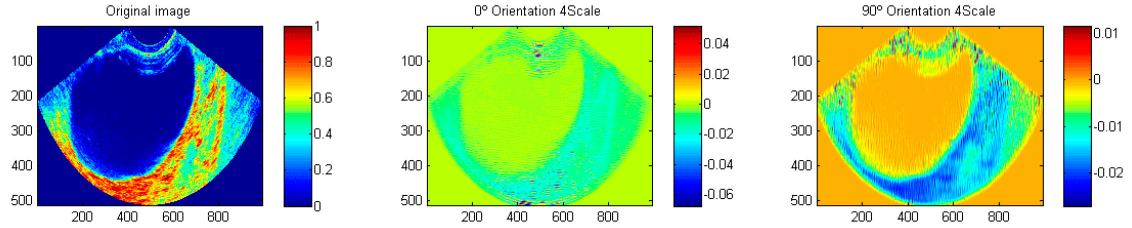
$$gI_{l,k} = I * gf_{(l,k)} \tag{4.17}$$

where * denotes the convolution operator. In this case, the biggest scale is 4 so the kernel size will be, at least 25. The feature vector is then constructed using the mean and standard deviation of the absolute value of the Gabor result as feature components. Therefore, the feature vector will have a length of 48: 24 means and 24 standard deviations for each image.

$$\mu_{l,k}(m,n) = \frac{1}{MxN}\sum_{m=1}^{M}\sum_{n=1}^{N}|gI_{l,k}(m,n)| \tag{4.18}$$

$$\sigma_{l,k}(m,n) = \sqrt{\frac{1}{MxN}\sum_{m=1}^{M}\sum_{n=1}^{N}(|gI_{l,k}(m,n)| - \mu_{l,k}(m,n))^2} \tag{4.19}$$

In Figure 4.20, a GW example and the original filters can be seen. In the example, the original image and two Gabor results are represented. The results correspond to the horizontal (0°) and vertical (90°) orientation and they are represented with pseudo-color in order to improve the details. Therefore, horizontal and vertical edges of the image are emphasized in those results. As the original image has a wide range of orientations, the mean parameter will have a low value.



	ORIGINAL	0ºOrientation 4Scale	90ºOrientation 4Scale
Mean	0.1790	0.0052	0.0047
Std	0.2591	0.0073	0.0061

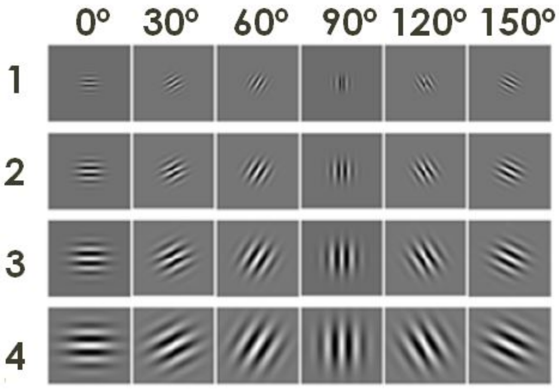


Figure 4.20: GW Example and GW filters

Chapter 5

Dimensionality Reduction

This chapter focus on the dimensionality reduction from the features. It can be defined as the process of reducing the number of variables under consideration, and can be divided into two subgroups, feature selection and feature extraction. In this way, we avoid working with features that are redundant which could affect our classifiers by leading to a dimensionality problem. Reducing features can also save storage and computation time.

5.1 Feature Data

After the feature calculation step, two variables are obtained:

- **Feature vector:** consists of 145 image observations divided into two groups, 106 benign and 39 malignant, with 591 features for each image. These are:
  - 512 LBP Histogram Counts: from (R=1; P=8) and (R=2; P=8)
  - 4 LBP Histogram Features: Variance and Skewness from (R=1; P=8) and (R=2; P=8)
  - 6 LBP Features: Entropy and Average Power from (R=1; P=8), (R=2; P=16), (R=3; P=24)
  - 1 FD
  - 2 Entropy (Shannon and Kapur)
  - 7 Invariant Moments
  - 3 GLCM: Correlation, Entropy and 4 Moment
  - 8 LTE
  - 48 GW: 24 GW Mean and 24 GW Std
- **Label:** defines the group to which the 145 image observations belong. 0 corresponds to the benign group and 1 to the malignant one.

In Figure 5.1, a diagram can be seen with the different steps of this chapter. First of all, an exploratory data analysis will be made. After that, three methods for dimensionality reduction. And finally, the data will be scaled as it will later be explained.

5.2 Exploratory Data Analysis

We start with an exploratory data analysis approach in order to summarize the main characteristics of our data through visual methods.

Therefore, in Figure 5.2 and 5.3 a high dimensional **scatter data visualization** of certain pair of variables will be represented in order to see their distribution and possible overlapping problems or redundant features. The benign group (0) will be represented with blue color, and the malignant one (1) with red. Besides, all of them will be rescaled to the range (0,1). We can conclude that those pair of features that represent a line distribution contribute less than the others to the classification, because they are more correlated. In the diagonal of the representation, the histogram of each feature can be seen. The information in both sides is symmetric so that we can focus on one of them.

The intensity of the variable pairs relation can be calculated through the **correlation coefficient (r)**. It measures the strength and the direction of a linear relationship between two variables and it is sometimes referred to as the Pearson product moment correlation coefficient. As a high number of variables is being considered, this coefficient will be represented in an image instead of with numeric values. The correlation can be in between $(-1, 1)$, so the absolute value will be represented, including both positive and negative correlations. The image will be square and its dimensions will be determined by the number of features. The diagonal will always represent the highest correlation, when comparing a feature to itself. A correlation greater than 0.8 is generally described as strong. Therefore, a binary image with just the correlation values that exceed 0.8 will be also represented, in order to highlight redundancy and facilitate the interpretation.

As we can see in Figure 5.5, the correlation matrix is represented for both SA and ROI procedures. The features are arranged as in the “5.1 Feature Data” section list. It can be seen that overall there are high correlated features. A zoom in the last 79 features is made, excluding the two histogram counts calculation, where the correlation of the other calculated features is represented. The same procedure is repeated for the ROI features. In this case, the correlation coefficient is smaller due to the fact that only the regions of interest are contained.

Moreover, the **silhouette value (si)** can be calculated for our data. This value is defined as a measure of how similar each point is to points in its own cluster, when compared to points in the other cluster. In this project, we associate cluster 1 to the benign and cluster 2 to the malignant group. This measure provides information about how different the two groups are for each feature, so that the significant ones can be detected. Besides, the Silhouette value can also provide information about the level of overlapping between both clusters. In the vertical axis, the 145 images are represented and the silhouette values in the horizontal one. The si value contrasts the average distance to elements in the same cluster with the average distance to elements in other clusters. Objects with a high si value are considered well clustered, objects with a low si value may be outliers. By default, the squared Euclidean distance between points is used. Inside each group or cluster, the images are reorder, from the maximum value to the minimum one. This silhouette value si is theoretically calculated, as:

$$si = \frac{(b_i - a_i)}{max(a_i, b_i)} \tag{5.1}$$

where a_i is the average distance from the ith point to the other points in the same cluster as i, and b_i is the minimum average distance from the ith point to points in a different cluster, minimized over clusters.

In Figure 5.6, the si values can be seen for the LBP Entropy 018 feature in the ROI procedure. The negative si values correspond to images which would have been more appropriate in its neighboring cluster. The near to zero si values correspond to images on the border of classification between the two clusters, benign and malignant one. Therefore, it is clear that we are dealing with complicated images. In this case, the malignant observations are correctly identified, but we have a high rate of false positives. Most of the benign observations are being considered as malignant. From these values, it can be concluded that the data has a high grade of overlapping.

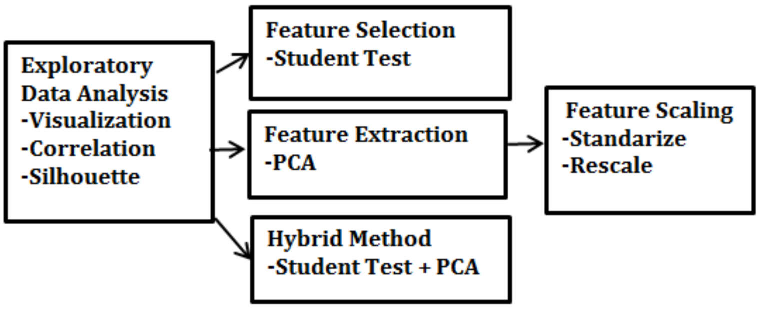


Figure 5.1: Diagram Dimension

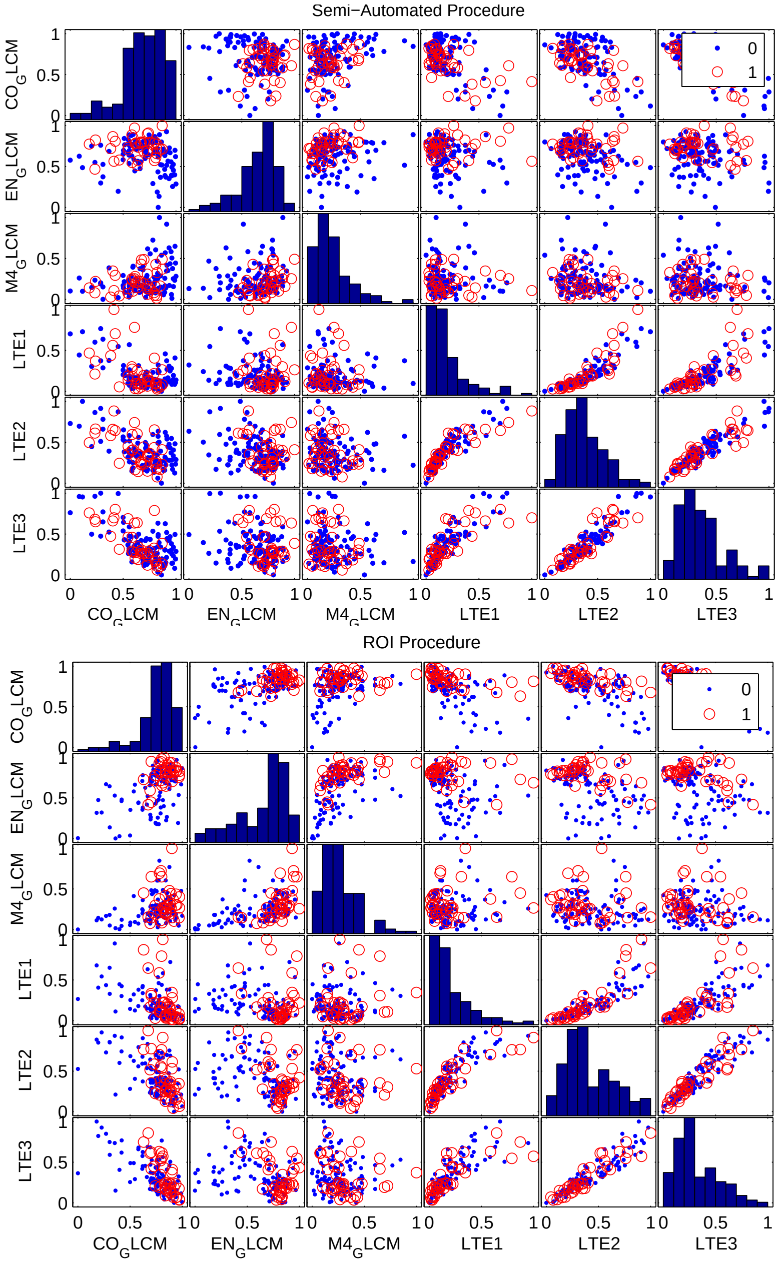


Figure 5.2: Scatter plot SA and ROI

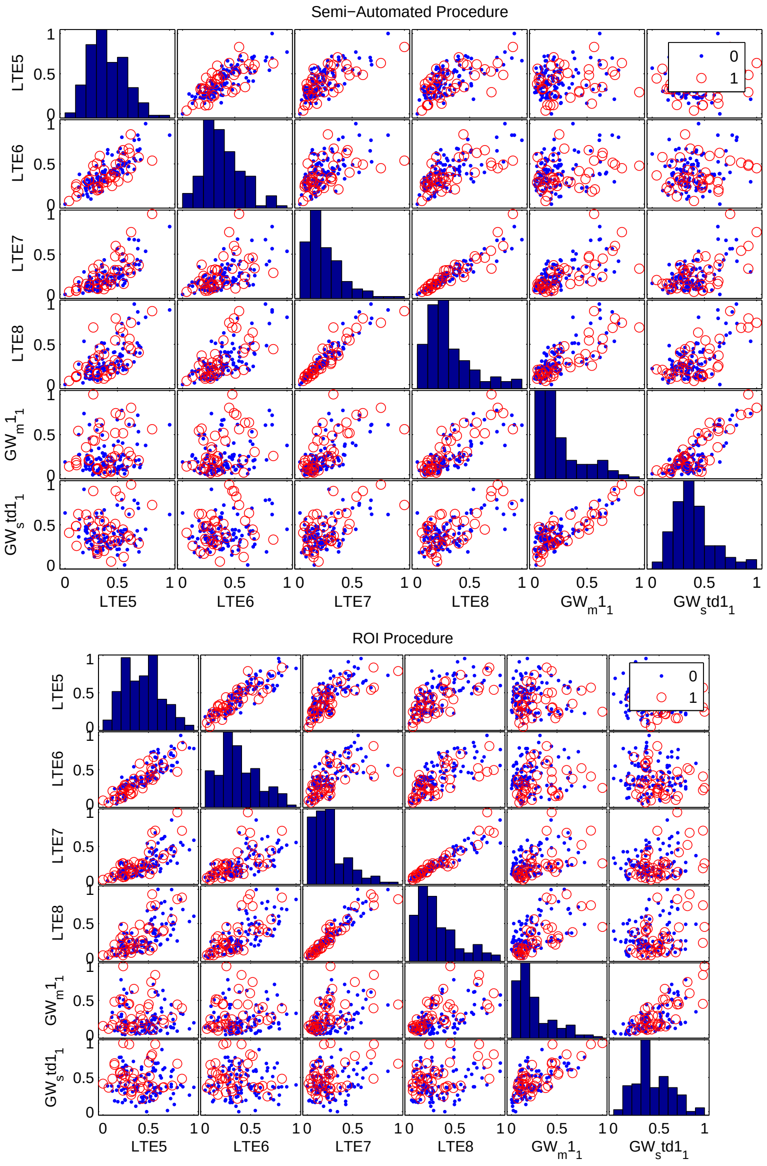


Figure 5.3: Scatter plot SA and ROI

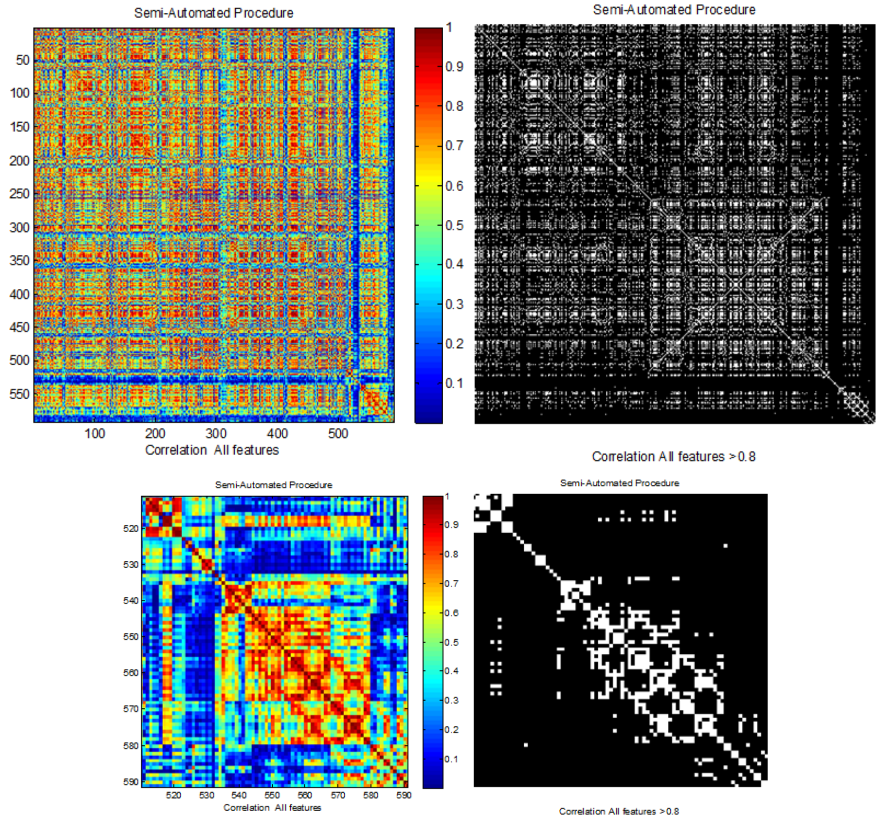


Figure 5.4: Correlation Matrix for Semi-automated Procedure

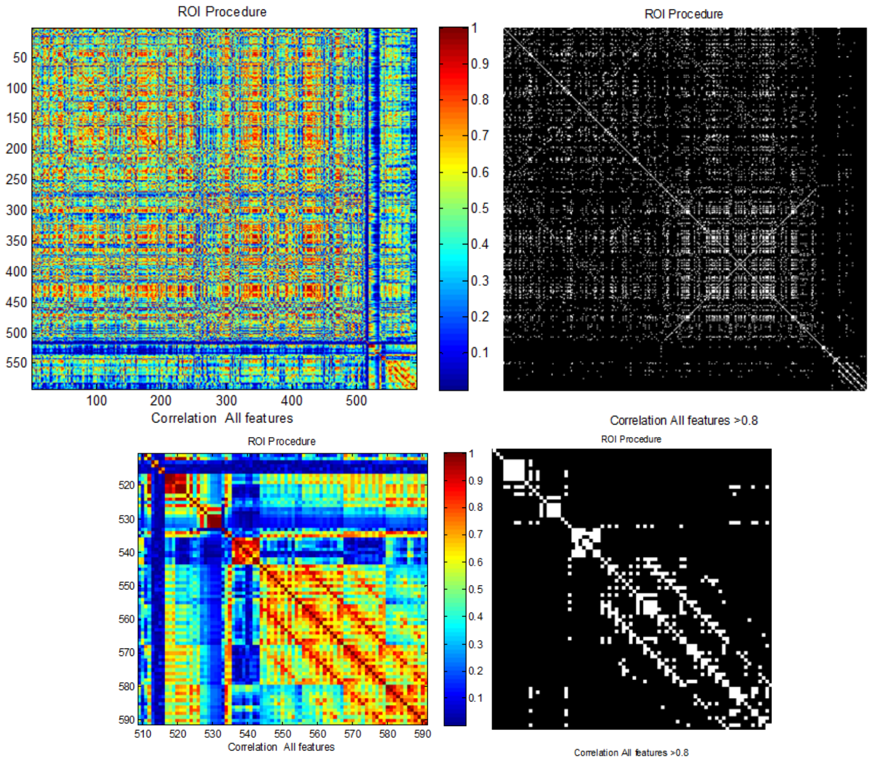


Figure 5.5: Correlation Matrix for ROI Procedure

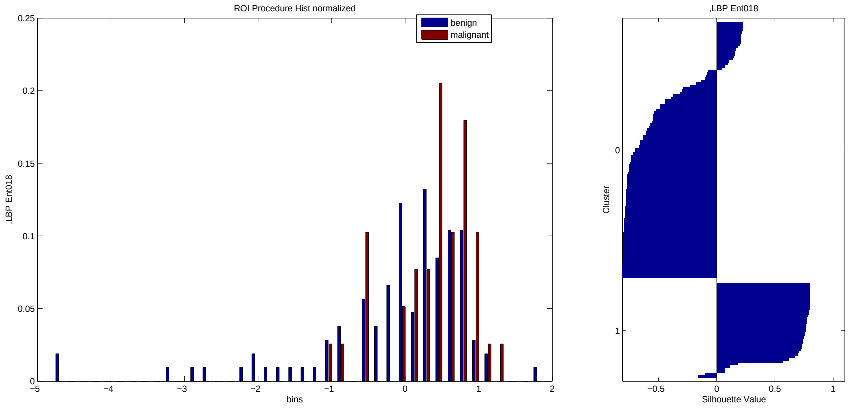


Figure 5.6: Histogram and Silhouette values for LBP Entropy 018 ROI Procedure

5.3 Feature selection

The feature selection approach tries to find a subset of the original variables, which are significant or relevant in order to use them in the model construction of the classifier. Redundant features will be those which provide no more information or repeated one. A possible algorithm is to test each possible subset of features finding the one which minimizes the error rate. Therefore, a t-test for two populations (benign and malignant group) will be applied on each feature and its p-value will be compared as a measure of how effective it is at separating groups, as in [7], [8], [9], [10]. In other words, with the Student test features with a statistically different mean for both groups can be selected.

5.3.1 Student t-test of 2 populations

The most common type of t-test, namely the Student t-test, is often used to assess whether the means of two classes are statistically different from each other by calculating a ratio between the difference of means and the variability of the two classes. There are several assumption underlying a t-test. These are:

- Each of the two populations being compared should follow a normal distribution. This can be tested using a normality test, or graphically using a normal quantile plot.
- The two populations being compared should have the same variance (testable using F-test, for example). However, if the size of the populations being compared is equal, the presence of unequal variances does not affect the test.

In probability theory [17], the **central limit theorem (CLT)** states that, given certain conditions, the arithmetic mean of a sufficiently large number of independent random variables (superior to 30) will be approximately normally distributed, regardless of the underlying distribution. In our case, the sample size contains a large number of image observations, specifically 106 benign and 39 malignant. Both groups exceed the limit of 30 samples, therefore we continue with the Student t-test implementation.

5.3.2 Results Semi-automated procedure

In Table 5.1, the significant features results can be seen. All of them fulfill to have a p-value less than 0.05, which indicates that the means are significantly different for the two classes: benign and malignant. 41 significant features have been selected. The others are not discriminating.

N	**Features**	**Benign**	**Malignant**	**p-value**
1	LBP hist Variance018	$3,87E-04 \pm 1,76E-04$	$3,20E-04 \pm 7,81E-05$	2,36E-02
2	LBP hist Variance028	$3,50E-04 \pm 2,01E-04$	$2,78E-04 \pm 5,52E-05$	2,81E-02
3	LBP hist Skewness018	$3,54E+00 \pm 1,42E+00$	$2,91E+00 \pm 4,08E-01$	7,14E-03
4	LBP hist Skewness028	$4,41E+00 \pm 1,64E+00$	$3,86E+00 \pm 8,57E-01$	4,87E-02
5	LBP Ent018	$4,611 \pm 0,240$	$4,714 \pm 4,714$	1,97E-02
6	LBP Ent216	$4,156 \pm 0,193$	$4,247 \pm 0,170$	9,77E-03
7	LBP AvgPower018	$0,652 \pm 0,030$	$0,637 \pm 0,007$	2,37E-03
8	LBP AvgPower216	$0,668 \pm 0,027$	$0,653 \pm 0,007$	1,57E-03
9	LBP AvgPower324	$0,664 \pm 0,028$	$0,649 \pm 0,008$	1,31E-03
10	E Shannon	$7,74 \pm 1,18E-01$	$7,37 \pm 1,58E-01$	7,68E-31
11	E Kapur	$7,74 \pm 1,15E-01$	$7,69 \pm 9,45E-2$	2,09E-02
12	IM1	$6,19E-01 \pm 1,82E-01$	$5,06E-01 \pm 1,13E-01$	3,79E-04
13	IM2	$9,71E-02 \pm 9,99E-02$	$5,41E-02 \pm 4,42E-02$	1,05E-02
14	GLCM CO	$9,35E-01 \pm 3,37E-02$	$9,17E-01 \pm 2,78E-02$	2,59E-03
15	GLCM EN	$2,40E+00 \pm 1,81E-01$	$2,50E+00 \pm 1,73E-01$	2,79E-03
16	GLCM M4	$3,01E-01 \pm 1,25E-01$	$3,92E-01 \pm 1,80E-01$	7,21E-04
17	GW mean 11	$8,86E-04 \pm 5,53E-04$	$1,18E-03 \pm 7,45E-04$	1,03E-02
18	GW mean 12	$8,88E-04 \pm 4,90E-04$	$1,10E-03 \pm 6,38E-04$	3,58E-02
19	GW mean 15	$5,79E-04 \pm 2,77E-04$	$6,93E-04 \pm 3,64E-04$	4,55E-02
20	GW mean 16	$8,85E-04 \pm 4,83E-04$	$1,12E-03 \pm 6,71E-04$	2,42E-02
21	GW mean 21	$1,89E-03 \pm 8,75E-04$	$2,41E-03 \pm 9,50E-04$	2,15E-03
22	GW mean 22	$1,77E-03 \pm 7,44E-04$	$2,11E-03 \pm 7,95E-04$	1,78E-02
23	GW mean 23	$1,10E-03 \pm 4,49E-04$	$1,28E-03 \pm 5,29E-04$	4,03E-02
24	GW mean 24	$7,18E-04 \pm 2,86E-04$	$8,32E-04 \pm 3,22E-04$	4,21E-02
25	GW mean 25	$1,09E-03 \pm 4,33E-04$	$1,26E-03 \pm 4,97E-04$	4,50E-02
26	GW mean 26	$1,77E-03 \pm 7,48E-04$	$2,13E-03 \pm 8,37E-04$	1,27E-02
27	GW mean 31	$3,16E-03 \pm 9,00E-04$	$3,79E-03 \pm 7,39E-04$	1,54E-04
28	GW mean 32	$2,86E-03 \pm 7,17E-04$	$3,25E-03 \pm 5,41E-04$	2,20E-03
29	GW mean 33	$1,80E-03 \pm 5,14E-04$	$2,04E-03 \pm 5,15E-04$	1,46E-02
30	GW mean 34	$1,36E-03 \pm 3,59E-04$	$1,53E-03 \pm 3,81E-04$	1,53E-02
31	GW mean 35	$1,79E-03 \pm 4,93E-04$	$2,02E-03 \pm 4,93E-04$	1,60E-02
32	GW mean 36	$2,85E-03 \pm 7,36E-04$	$3,29E-03 \pm 6,00E-04$	1,00E-03
33	GW mean 41	$6,53E-03 \pm 1,15E-03$	$7,19E-03 \pm 9,73E-04$	1,86E-03
34	GW mean 42	$3,60E-03 \pm 6,53E-04$	$3,97E-03 \pm 4,23E-04$	1,34E-03
35	GW mean 43	$2,45E-03 \pm 5,01E-04$	$2,69E-03 \pm 4,22E-04$	6,99E-03
36	GW mean44	$5,53E-03 \pm 1,03E-03$	$6,06E-03 \pm 9,90E-04$	6,15E-03
37	GW mean45	$2,43E-03 \pm 4,76E-04$	$2,67E-03 \pm 4,22E-04$	5,52E-03
38	GW mean46	$3,60E-03 \pm 6,64E-04$	$3,99E-03 \pm 4,43E-04$	8,10E-04
39	GW std 21	$2,91E-03 \pm 8,95E-04$	$3,30E-03 \pm 1,06E-03$	2,72E-02
40	GW std 31	$4,02E-03 \pm 8,43E-04$	$4,47E-03 \pm 7,84E-04$	4,21E-03
41	GW std 41	$6,73E-03 \pm 7,33E-04$	$7,05E-03 \pm 6,97E-04$	1,93E-02

Table 5.1: Student t-test significant results ($p\text{-value} < 0.05$). Semi-automated procedure

5.3.3 Results ROI procedure

In Table 5.2, the significant features results for the ROI procedure can be seen with a pvalue less than 0.05. A total of 58 significant features are selected. In comparison with the SA procedure, there are more significant features due to the fact that just the area of interest is taken into account.

N	**Features**	**Benign**	**Malignant**	**p-value**
1	LBP Ent018	$4,834 \pm 0,5840$	$5,120 \pm 0,294$	0,0041
2	LBP Ent216	$4,335 \pm 0,523$	$4,601 \pm 0,247$	2,73E-03
3	LBP Ent324	$3,659 \pm 0,480$	$3,898 \pm 0,330$	0,004
4	LBP AvgPower018	$4,834 \pm 0,584$	$5,120 \pm 0,294$	4,14E-03
5	LBP AvgPower216	$9,673 \pm 0,547$	$9,447 \pm 0,313$	0,016
6	LBP AvgPower324	$4,335 \pm 0,523$	$4,601 \pm 0,247$	2,73E-03
7	E Shannon	$6,30 \pm 1,12$	$7,08 \pm 3,06E-01$	3,58E-05
8	E Kapur	$7,23E+00 \pm 3,47E-01$	$7,48E+00 \pm 1,57E-01$	2,74E-05
9	IM1	$1,223 \pm 1,166$	$0,585 \pm 0,262$	9,38E-04
10	GLCM EN	$1,83E+00 \pm 5,60E-01$	$2,31E+00 \pm 2,73E-01$	1,03E-06
11	GLCM M4	$2,77E-01 \pm 1,60E-01$	$4,08E-01 \pm 2,04E-01$	7,95E-05
12	LTE 6	$3,993 \pm 1,094$	$3,581 \pm 1,037$	0,043
13	GW mean 11	$1,38E-03 \pm 8,90E-04$	$1,84E-03 \pm 1,15E-03$	1,15E-02
14	GW mean 12	$1,04E-03 \pm 6,14E-04$	$1,41E-03 \pm 8,30E-04$	4,83E-03
15	GW mean 13	$7,03E-04 \pm 3,53E-04$	$9,15E-04 \pm 4,62E-04$	3,96E-03
16	GW mean 14	$5,88E-04 \pm 2,76E-04$	$7,12E-04 \pm 2,87E-04$	1,89E-02
17	GW mean 15	$7,24E-04 \pm 3,66E-04$	$8,83E-04 \pm 3,86E-04$	2,35E-02
18	GW mean 16	$1,09E-03 \pm 6,60E-04$	$1,40E-03 \pm 8,85E-04$	2,25E-02
19	GW mean 21	$2,73E-03 \pm 1,44E-03$	$3,52E-03 \pm 1,47E-03$	4,15E-03
20	GW mean 22	$2,02E-03 \pm 1,02E-03$	$2,65E-03 \pm 1,10E-03$	1,57E-03
21	GW mean 23	$1,24E-03 \pm 5,98E-04$	$1,61E-03 \pm 6,40E-04$	1,56E-03
22	GW mean 24	$9,74E-04 \pm 4,59E-04$	$1,19E-03 \pm 4,43E-04$	1,34E-02
23	GW mean 25	$1,28E-03 \pm 5,97E-04$	$1,55E-03 \pm 5,89E-04$	1,54E-02
24	GW mean 26	$2,10E-03 \pm 1,07E-03$	$2,61E-03 \pm 1,14E-03$	1,46E-02
25	GW mean 31	$4,25E-03 \pm 1,69E-03$	$5,24E-03 \pm 1,21E-03$	9,61E-04
26	GW mean 32	$3,19E-03 \pm 1,21E-03$	$4,01E-03 \pm 9,47E-04$	1,91E-04
27	GW mean 33	$2,01E-03 \pm 7,97E-04$	$2,51E-03 \pm 6,81E-04$	6,40E-04
28	GW mean 34	$1,65E-03 \pm 6,44E-04$	$2,01E-03 \pm 5,44E-04$	2,29E-03
29	GW mean 35	$2,05E-03 \pm 7,98E-04$	$2,46E-03 \pm 6,86E-04$	4,92E-03
30	GW mean 36	$3,29E-03 \pm 1,28E-03$	$3,96E-03 \pm 9,72E-04$	3,36E-03
31	GW mean 41	$6,46E-03 \pm 2,57E-03$	$8,39E-03 \pm 1,72E-03$	2,67E-05
32	GW mean 42	$3,97E-03 \pm 1,34E-03$	$4,86E-03 \pm 8,08E-04$	1,52E-04
33	GW mean 43	$2,74E-03 \pm 9,28E-04$	$3,31E-03 \pm 6,31E-04$	5,31E-04
34	GW mean 44	$4,79E-03 \pm 2,11E-03$	$6,60E-03 \pm 1,66E-03$	3,37E-06
35	GW mean 45	$2,76E-03 \pm 9,30E-04$	$3,27E-03 \pm 7,20E-04$	2,60E-03
36	GW mean 46	$4,05E-03 \pm 1,39E-03$	$4,78E-03 \pm 8,48E-04$	2,69E-03
37	GW std 11	$2,54E-03 \pm 8,84E-04$	$2,99E-03 \pm 9,35E-04$	8,81E-03
38	GW std 12	$2,03E-03 \pm 6,31E-04$	$2,31E-03 \pm 7,70E-04$	2,63E-02
39	GW std 13	$1,71E-03 \pm 4,75E-04$	$1,94E-03 \pm 6,05E-04$	1,81E-02
40	GW std 14	$1,55E-03 \pm 4,83E-04$	$1,74E-03 \pm 5,46E-04$	4,80E-02
41	GW std 15	$1,73E-03 \pm 5,22E-04$	$1,97E-03 \pm 4,86E-04$	1,28E-02
42	GW std 21	$3,27E-03 \pm 1,08E-03$	$3,84E-03 \pm 1,12E-03$	6,04E-03
43	GW std 22	$2,65E-03 \pm 9,20E-04$	$3,12E-03 \pm 1,05E-03$	1,03E-02
44	GW std 23	$1,99E-03 \pm 6,22E-04$	$2,34E-03 \pm 7,74E-04$	4,98E-03
45	GW std 24	$1,65E-03 \pm 4,96E-04$	$1,87E-03 \pm 5,79E-04$	2,91E-02
46	GW std 25	$2,01E-03 \pm 6,50E-04$	$2,31E-03 \pm 6,23E-04$	1,53E-02
47	GW std 31	$4,25E-03 \pm 1,11E-03$	$4,87E-03 \pm 9,25E-04$	2,14E-03

N	**Features**	**Benign**	**Malignant**	**p-value**
48	GW std 32	$3,48E-03 \pm 9,53E-04$	$4,04E-03 \pm 8,73E-04$	1,69E-03
49	GW std 33	$2,50E-03 \pm 7,11E-04$	$2,93E-03 \pm 7,39E-04$	1,68E-03
50	GW std 34	$2,03E-03 \pm 5,66E-04$	$2,29E-03 \pm 6,19E-04$	1,57E-02
51	GW std 35	$2,52E-03 \pm 7,47E-04$	$2,88E-03 \pm 6,65E-04$	1,02E-02
52	GW std 36	$3,53E-03 \pm 1,02E-03$	$3,98E-03 \pm 9,68E-04$	1,82E-02
53	GW std 41	$5,62E-03 \pm 1,43E-03$	$6,79E-03 \pm 9,87E-04$	5,59E-06
54	GW std 42	$3,94E-03 \pm 8,66E-04$	$4,52E-03 \pm 6,87E-04$	2,55E-04
55	GW std 43	$2,92E-03 \pm 7,24E-04$	$3,35E-03 \pm 6,30E-04$	1,36E-03
56	GW std 44	$3,77E-03 \pm 1,11E-03$	$4,72E-03 \pm 8,75E-04$	3,50E-06
57	GW std 45	$4,05E-03 \pm 1,11E-03$	$4,65E-03 \pm 9,27E-04$	3,30E-03
58	GW std 46	$2,93E-03 \pm 7,42E-04$	$3,29E-03 \pm 6,76E-04$	9,74E-03

Table 5.2: Student t-test significant results (p-value$<$ 0.05). ROI procedure

5.4 Feature extraction

Feature extraction is another technique for dimensionality reduction. It is specially efficient with data that is not only large but also redundant. As we have seen in the previous section, our data is correlated so it could be interesting to transform it into a reduced representation set of features. It is expected that the extracted features set will extract the relevant information. In our work, the principal component analysis technique will be applied.

5.4.1 Principal Component Analysis (PCA)

Principal component analysis is a statistical procedure. It is based on an orthogonal transformation that converts a set of image observations with correlated feature variables into a set of uncorrelated variables, called principal components. Therefore, the number of principal components is always less than or equal to the number of image observations available minus one. This is because the covariance matrix has a size corresponding to the number of observations minus one when doing PCA on centered data. One image is treated as a single point in a high-dimensional space. The orthogonal transformation is defined so that the first principal component has the largest possible variance, in a way that represents as much variability as possible. These components are orthogonal because they correspond to the eigenvectors of the covariance matrix, which is symmetric.

In this way, PCA transforms a set of correlated variables into a new set of uncorrelated variables. If the original variables were almost uncorrelated, the gain of the transformation will not be high. Therefore, in our case this method can be profitable. Furthermore, it is important to take into account that PCA is sensitive to the relative scaling of the original variables, so the entire data needs to be in the same scale. Our features do not have the same units, because not all of them are comparable: bits of information, pixel counts, a complex number and so on. Therefore, the data will be first standardized, and then the PCA analysis will be applied. In comparison with the feature selection approach, a big reduction in the number of features can be seen. The maximal variability will be explained by the whole number of observations minus one, i.e. 144. However, in comparison with the Student Test selection, in this case we are using all the features, including LBP histogram counts, as part of the input data.

5.4.2 Results Semi-automated procedure

In Figure 5.7, two figures that represent the components variability for the SA procedure can be seen. The one on the left represents the amount of variance explained for the first ten principal components. With 7 of the 591 features we can already explain the 90% of the variability, although the first component is the ones that more variability explains. The image on the right represents the number of coefficients needed to explain a higher percentage of variability. Therefore, it can be seen that :

- To explain 90%, we need 7 coefficients.
- To explain 95%, we need 16 coefficients.
- To explain 99%, we need 70 coefficients.
- To explain 100%, we need 144 coefficients.

As we have said before, these percentages are associated with the fact that the variables are highly correlated. The more coefficients we take, the more variability we can explain, but also more computation is needed.

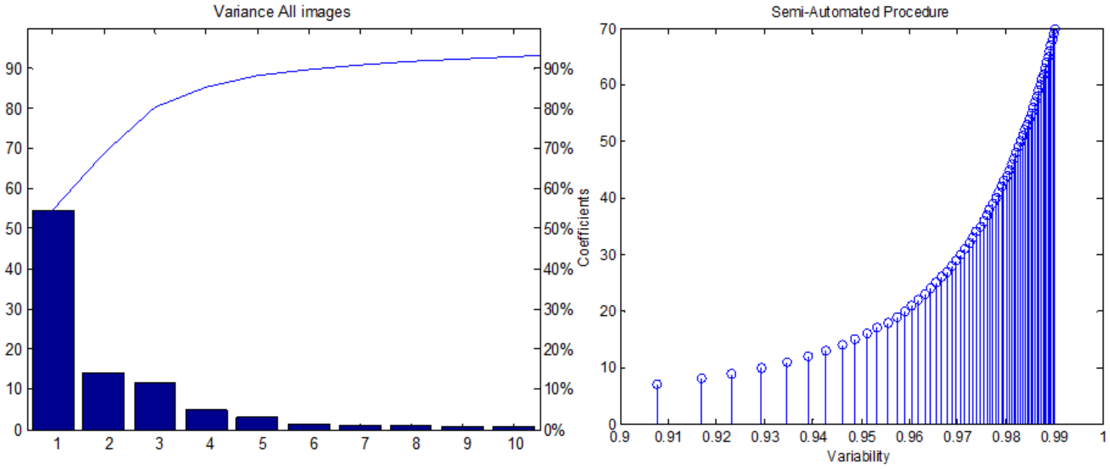


Figure 5.7: PCA Variability. Semi-automated Procedure

If we focus on the first coefficient, the one that represents the main part of the variability, it is possible to see which is the combination of features presented on it. In the following expression, only the 20 features with higher weight on the first coefficient will be represented due to the fact that we are working with a large number of features. Those are:

$$\begin{aligned} COEF1 = {} & 0,0546 \cdot \text{LBPhist028 174} + 0,0545 \cdot \text{LBPhist028 219} + 0,0545 \cdot \text{LBPhist028 183} \\ & + 0,0545 \cdot \text{LBPhist028 108} + 0,0542 \cdot \text{LBPhist028 74} + 0,0603 \cdot \text{LBPhist028 83} \\ & + 0,0542 \cdot \text{LBPhist028 167} + 0,0542 \cdot \text{LBPhist028 38} + 0,0602 \cdot \text{LBPhist028 170} \\ & + 0,0541 \cdot \text{LBPhist028 44} + 0,0540 \cdot \text{LBPhist028 149} + 0,0601 \cdot \text{LBPhist028 173} \\ & + 0,0540 \cdot \text{LBPhist028 107} + 0,0540 \cdot \text{LBPhist018 95} + 0,0540 \cdot \text{LBPhist028 78} \\ & + 0,0539 \cdot \text{LBPhist028 179} + 0,0539 \cdot \text{LBPhist028 203} + 0,0598 \cdot \text{LBPhist018 245} \\ & + 0,0539 \cdot \text{LBPhist028 155} \end{aligned}$$

It can clearly be seen that the LBP histograms 018 and 028 express a high percentage of variability in our dataset. It was supposed, as explained in the chapter before, to be a significant feature. Therefore, it was directly included in the feature selection process without applying the student test to it. If the LBP histogram counts are removed, it is possible to see the importance of the other features, which also explain an important part of the variability. This importance is the following:

$$\begin{aligned} COEF1 = {} & 0,0510 \cdot \mathrm{GW\ m1\ 6} + 0,0509 \cdot \mathrm{GW\ m1\ 2} + 0,0501 \cdot \mathrm{GW\ m1\ 5} \\ & + 0,0499 \cdot \mathrm{LBP\ Ent324} + 0,0543 \cdot \mathrm{GW\ m2\ 2} + 0,0493 \cdot \mathrm{GW\ std2\ 6} \\ & + 0,0493 \cdot \mathrm{GW\ m1\ 3} + 0,0489 \cdot \mathrm{GW\ m2\ 6} + 0,0489 \cdot \mathrm{GW\ std2\ 2} \\ & + 0,0488 \cdot \mathrm{LTE8} + 0,0488 \cdot \mathrm{LTE7} + 0,0487 \cdot \mathrm{GW\ m2\ 5} \\ & + 0,0482 \cdot \mathrm{GW\ m2\ 3} + 0,0476 \cdot \mathrm{GW\ m1\ 1} + 0,0472 \cdot \mathrm{LTE1} \\ & + 0,0465 \cdot \mathrm{GW\ std2\ 1} + 0,0465 \cdot \mathrm{GW\ std2\ 3} + 0,0464 \cdot \mathrm{LTE2} \\ & + 0,0463 \cdot \mathrm{GW\ std2\ 5} \end{aligned}$$

The Gabor Wavelet feature, for example, seems to have importance in the first principal coefficient. This coefficient does not represent all features similarly, there are also some other features (about 89) with a negative sign on it. This fact is also represented in Figure 5.8, where the factorial plane (first and second coefficients) is plotted. It can be seen, that both the first (horizontal axis) and second (vertical axis) coefficients have positive and negative variables. In the first component, unlike in the second one, it is possible to differentiate between two groups of variables. This group differentiation can lead to a possible discrimination in the classifier. The variables that are close to zero are not explained. The red points of the diagram indicate with coordinates the score of each observation for the two principal components. In Figure 5.9, the first three principal components can be seen.

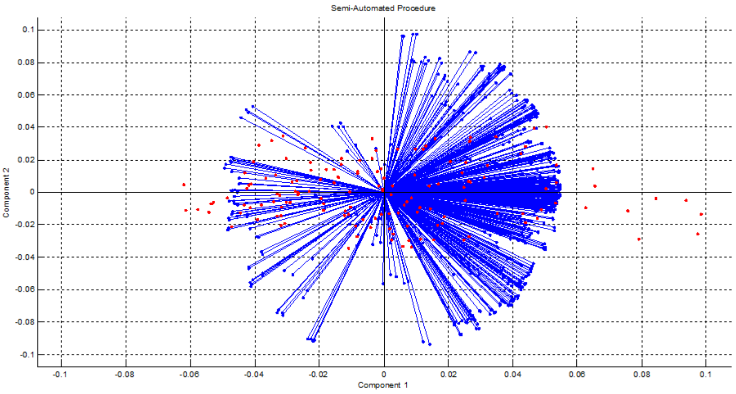


Figure 5.8: PCA 1 and 2 components. Semi-automated Procedure

5.4.3 Results ROI procedure

In Figure 5.10, two figures that represent the components variability for the ROI procedure can be seen. In this case, the left one determines the explained variability of the ten principal components. Whereas the right one determines the number of coefficients needed to explain a higher variability. It can be seen that, in comparison with SA, a higher number of coefficients is needed to explain the same variability of the data. Specifically:

- To explain 90%, we need 11 coefficients.
- To explain 95%, we need 27 coefficients.
- To explain 99%, we need 83 coefficients.
- To explain 100%, we need 144 coefficients.

Therefore, 145 images with 591 features can be represented in 144 different planes. To conclude, it can be said that in the ROI procedure more PCA coefficients are needed, because the data is less redundant.

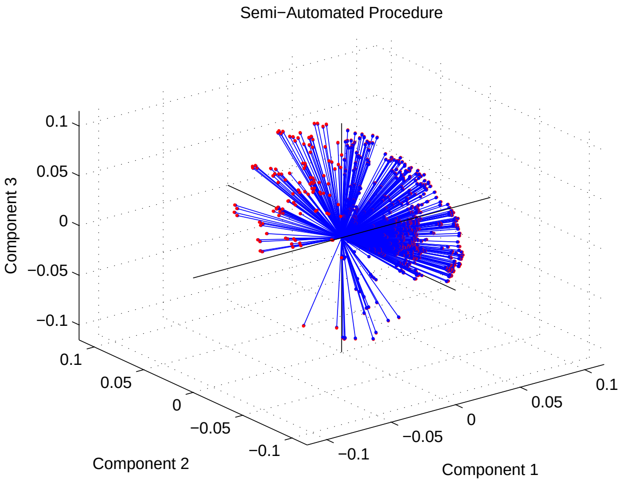


Figure 5.9: PCA three components. Semi-automated Procedure

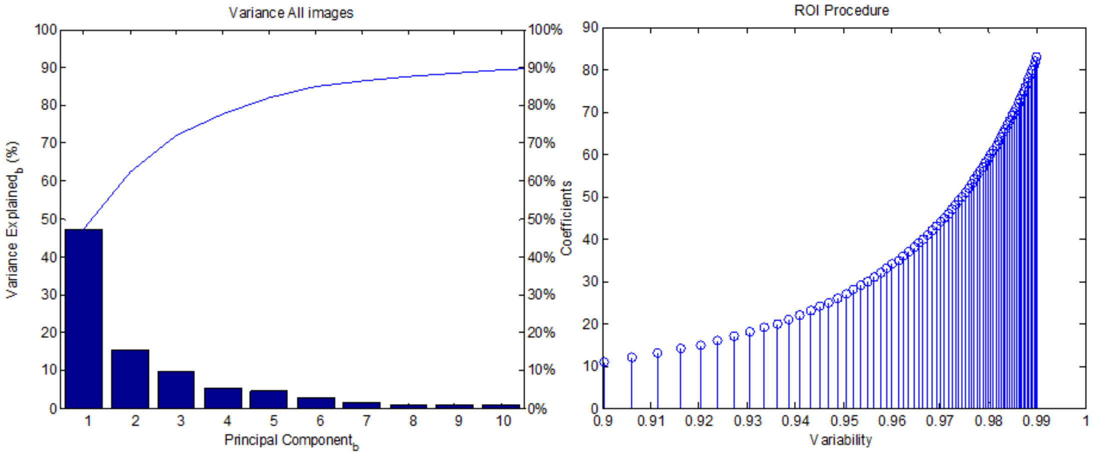


Figure 5.10: PCA Variability. ROI Procedure

In this case, if we also focus our interest on the first coefficient, we can see which is the combination of features presented on it. We will also represent the 20 features with more weight on this first coefficient. These are:

$$\begin{aligned}COEF1 = {} & 0,0574 \cdot \text{LBPhist028 44} + 0,0574 \cdot \text{LBPhist028 213} + 0,0571 \cdot \text{LBPhist028 174} \\ & + 0,0571 \cdot \text{LBPhist028 183} + 0,0571 \cdot \text{LBPhist028 170} + 0,0624 \cdot \text{LBPhist028 213} \\ & + 0,0571 \cdot \text{LBPhist028 105} + 0,0570 \cdot \text{LBPhist018 256} + 0,0570 \cdot \text{LBPhist028 175} \\ & + 0,0569 \cdot \text{LBPhist028 172} + 0,0569 \cdot \text{LBPhist028 122} + 0,0568 \cdot \text{LBPhist028 235} \\ & + 0,0568 \cdot \text{LBPhist028 45} + 0,0568 \cdot \text{LBPhist028 174} + 0,0568 \cdot \text{LBPhist028 212} \\ & + 0,0567 \cdot \text{LBPhist018 47} + 0,0567 \cdot \text{LBPhist028 152} + 0,0567 \cdot \text{LBPhist028 155} \\ & + 0,0566 \cdot \text{LBPhist018 227}\end{aligned}$$

The LBP histograms 018 and 028 have also high significance in this ROI procedure. Removing the histogram bins, the following features can be seen.

$$\begin{aligned}COEF1 = {} & 0,0485 \cdot \text{GW m1 3} + 0,0483 \cdot \text{GW m2 6} + 0,0478 \cdot \text{LTE7} \\ & + 0,0473 \cdot \text{IM4} + 0,0467 \cdot \text{GW m1 5} + 0,0467 \cdot \text{LBP Ener018} \\ & + 0,0467 \cdot \text{LBP Ener216} + 0,0464 \cdot \text{IM4} + 0,0464 \cdot \text{IM1} \\ & + 0,0463 \cdot \text{IM3} + 0,0463 \cdot \text{LTE5} + 0,0461 \cdot \text{LTE4} \\ & + 0,0461 \cdot \text{LTE1} + 0,0038 \cdot \text{LTE7} + 0,0460 \cdot \text{LBPhist Skew} \\ & + 0,0460 \cdot \text{GLCM CO} + 0,0031 \cdot \text{GW std1 4} + 0,0458 \cdot \text{LTE2} \\ & + 0,0458 \cdot \text{LTE8} + 0,0457 \cdot \text{LTE6}\end{aligned}$$

Some Gabor Wavelet features, invariant moments and LBP Energies seem to have importance for the first principal coefficient. This fact is also represented in Figure 5.11, where the factorial plane (first and second coefficients) is plotted. The first (horizontal axis) and second (vertical axis) coefficients have positive and negative variables. In this case, it can also be seen that with the first component, it is easier to differentiate between two groups of variables. The red points also indicate the score of each observation. In Figure 5.12, the first three principal components can be seen.

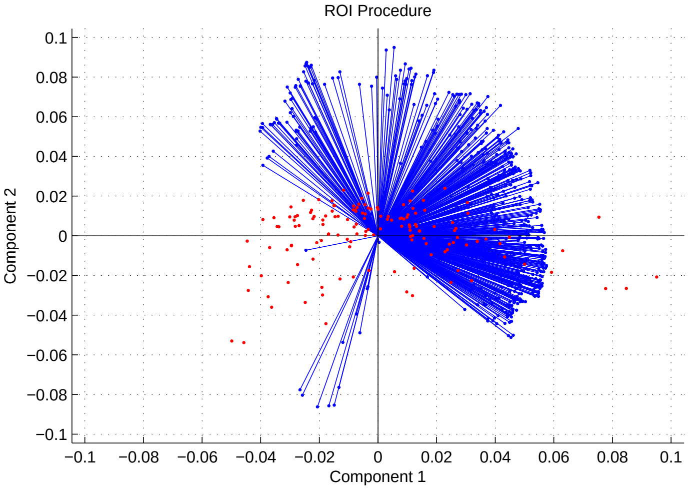


Figure 5.11: PCA 1 and 2 components. ROI Procedure

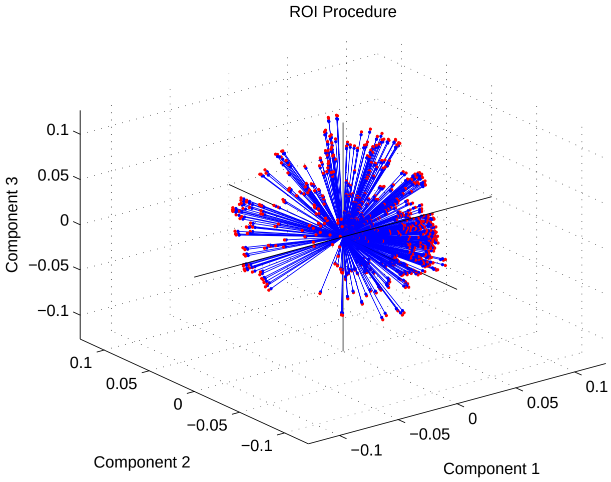


Figure 5.12: PCA three components. ROI Procedure

5.5 Hybrid Method: ST and PCA

Another possible approach will be a combination of both Student Test results and Principal Components Analysis. Both methods have their own advantages and disadvantages. In Figure 5.13, the worst cases of each method can be seen. The left one corresponds to ST method and the right one to PCA.

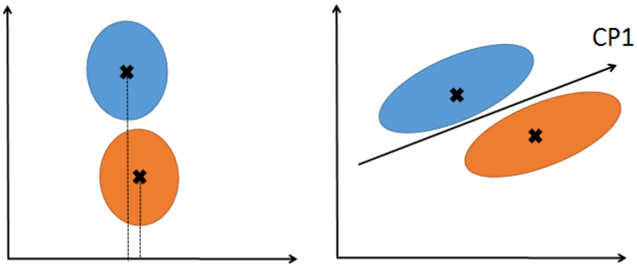


Figure 5.13: Problems in ST and PCA

As we can see on the left image, two groups can be differentiated. However, their mean values are not significantly different because they are very close to each other, so it will not be selected. On the right, two groups can also be differentiated. However, the first principal component will indicate the direction with the highest variability. If we consider just this component supposing that the explained percentage is enough, it will not be possible to differentiate them.

If the methods are now exchanged, the problem will disappear, as it can be seen in Figure 5.14. On the left, the groups with no mean significance can be separated with the first principal component. On the right, the groups with high variability can also be significant due to the mean difference.

Therefore, a combination of both methods creating an hybrid one will lead to a proper selection of features and a better dimensional reduction.

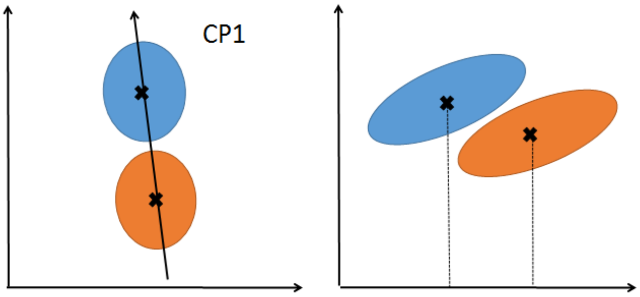


Figure 5.14: Solution in ST and PCA

5.6 Feature scaling

Feature scaling or feature normalization is a method used to standardize the range of independent variables or features of data. Since the range of values of raw data varies widely, there is a need to normalize.

First of all, it is applied after the Student-test and before PCA. It is needed that the features from PCA have similar variances, if not the feature with biggest variance will control the first principal component. During the Student test, feature scaling is not needed because each feature is compared individually. However, it will be needed before the classifier, as it will be later explained.

Two main methods can be differentiated:

1. **Rescaling**: Consists in a rescale of the range of features in order to make them independent. We will use the range [0,1] for scaling. The value will be given by the following equation, where x is the original value, x' is the normalized one.

$$x' = \frac{x - min(x)}{max(x) - min(x)} \tag{5.2}$$

2. **Standarization**: It makes the values of each feature in the data have zero-mean and unit-variance. The general method is to determine the distribution mean and standard deviation for each feature. Next we subtract the mean and we divide the values of each feature by its standard deviation.

$$x' = \frac{x - mean(x)}{std(x)} \tag{5.3}$$

We will use both of them through the thesis.

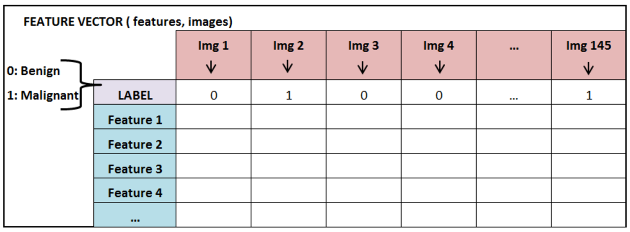
5.7 Feature vector Summary (FV)

From the original 591 calculated features, we have reduced their dimensionality in both the semi automated procedure and in the ROI one. In Figure 5.15, the FV structure can be seen.

5.7.1 FV Semiautomated Procedure

The Semiautomated procedure differentiates between:

- **Feature selection (ST):** We have chosen 41 significant features, which have a p-value less than 0.05. To all these features, we add the LBP histogram counts that are supposed to be already significant. Therefore, plus 510 counts, we have 551 features.
- **Feature extraction (PCA):** To explain a 95% of the variability, we need 21 coefficients.



FEATURE VECTOR (features, images)

0: Benign

1: Malignant

	Img 1 ↓	Img 2 ↓	Img 3 ↓	Img 4 ↓	...	Img 145 ↓
LABEL	0	1	0	0	...	1
Feature 1						
Feature 2						
Feature 3						
Feature 4						
...						

Figure 5.15: Feature Vector

5.7.2 FV ROI Procedure

The ROI procedure differentiates between:

- **Feature selection (ST):** We have chosen 58 significant features, which have a p-value less than 0.05. To all these features, we add the LBP histogram counts that are supposed to be already significant. Therefore, plus 510 counts, we have 568 features.
- **Feature extraction (PCA):** To explain a 95% of the variability, we need 37 coefficients.

Chapter 6

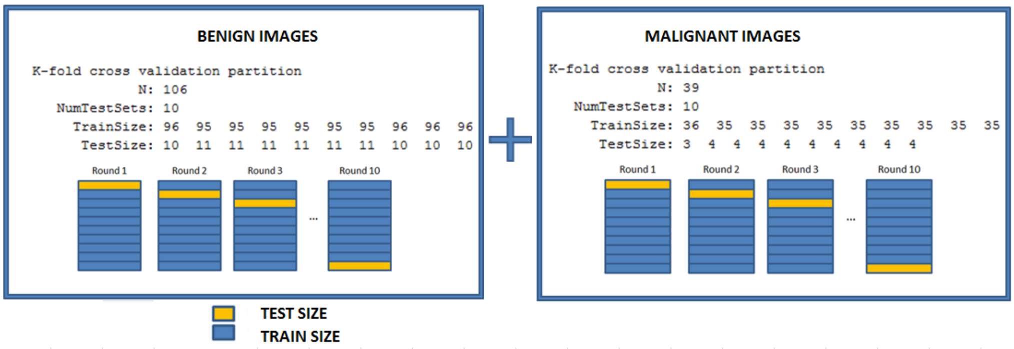
Classification System

This chapter will lead with the implementation of an accurate classifier, called Neural Networks. Different configurations will be made and all the results compared.

6.1 Classifier Evaluation: K-fold cross validation (CV)

The feature vector observations will be used to train our classifier. In order to have an indication of how good is our classifier when dealing with new predictions, the entire data will not be used when training it. The removed data will be used to test the performance of our classifier. Therefore, data needs to be divided into two sets: training data and test data. The training data will train the classifier and the test data (remaining samples) will be used to evaluate its performance. This evaluation method is called, **cross-validation**.

One kind of cross validation is called, **K-fold cross validation**. As we can see in Figure 6.1, the data set is divided into $k_i (i = 1, 2, ..., K)$ disjoint subsets. Each time, one of the k_i subsets is used as the test set and the other subsets are collect to form a training set. Then, the average error or validation accuracy across all subsets is computed in order to get a final cross-validation accuracy. $K = 10$ different sets divisions will be considered. Each sub-sample has roughly equal size. As the same benign and malignant image proportion as in our database wants to be ensured, a separate cross validation will be made for both groups (106 for benign and 39 for malignant). Then, the corresponding train and test sets will be joined. The training and testing size tend to have the same size in all the divisions. Different executions of the cross validation code will lead to different data divisions.



```
K-fold cross validation partition
             N: 106
   NumTestSets: 10
     TrainSize: 96   95   95   95   95   95   95   96   96   96
      TestSize: 10   11   11   11   11   11   11   10   10   10
```

```
K-fold cross validation partition
             N: 39
   NumTestSets: 10
     TrainSize: 36   35   35   35   35   35   35   35   35   35
      TestSize: 3   4   4   4   4   4   4   4   4   4
```

Figure 6.1: Cross validation example

The advantage of this method is that it does not depend on how the data gets divided. Every data point gets to be in a test set exactly once, and gets to be in a training set $K-1$ times. Besides, it is ensured that all the test sets are completely different from one to the other, avoiding repetitions. Therefore, the variance of the resulting estimate is reduced as the number of sets K is increased. However, the training algorithm will have to be rerun from scratch more times and it will involve more computation.

6.2 Artificial Neural Network Classifier (ANN)

The name neural networks covers a wide variety of processing architectures which involve simple processing units with a large number of connections by weighted links. The basic idea takes inspiration from models in true neurons, therefore it is called artificial neural networks. In Figure 6.2, we can see the relationship between the neural and the artificial model, where the axon is related to the output, the soma is the activation function and the synapses are the weighted inputs. The neural network function starts with several inputs (plus a bias term). Then, it multiplies each input by its correspondent weight. Inside the neuron, an activation function is applied to the sum of the results, and finally an output result is obtained.

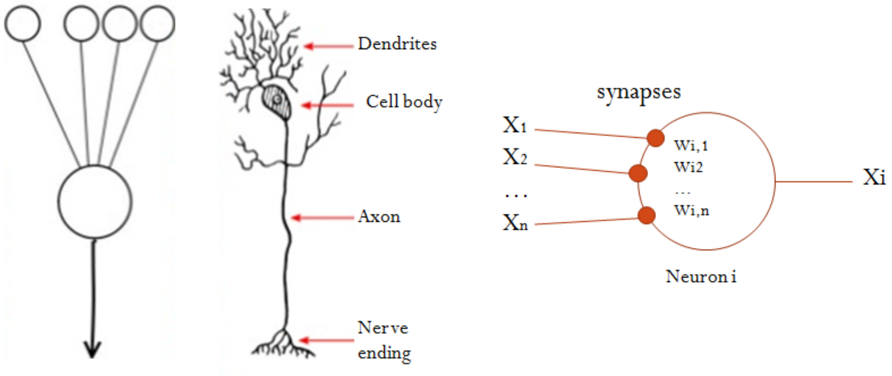


Figure 6.2: Neural Network Relationship

A single neuron cannot do very much. However, several neurons can be combined into a layer or multiple layers that have great power. Therefore, a neural network model is built from many neurons, where each neuron contains a learning model. The neuron (also called activation unit) has features as input, and the output is the **model** $\mathbf{h(x)}$. The layers in between are called Hidden Layers.

Each unit in the hidden layer is a weighted sum of the values in the first layer. Bias and Weights neurons can also be differentiated. Bias neurons can shift the transfer function curve horizontally allowing us to customize the input-to-output mapping to suit our particular needs. Weights neurons, also called thetas or parameters, manipulate the shape or curvature of the transfer function. An example of a real artificial network is the perceptron. In Figure 6.3, an example of the weights and bias role is shown in a perceptron network with a sigmoid activation function. A small weight (red) is related to a gradual slope, whereas a high weight (blue) is related to a steep one.

One major advantage, is that we are not constrained to the basic input features. Considering a three layers network, the raw features are just the input for the second (hidden) layer. Then, the hidden layer learns those features and finds its theta parameters. Therefore, the output layer will not be using the raw input features, but the hidden layer learned ones. Typically, an ANN is defined by three types of parameters:

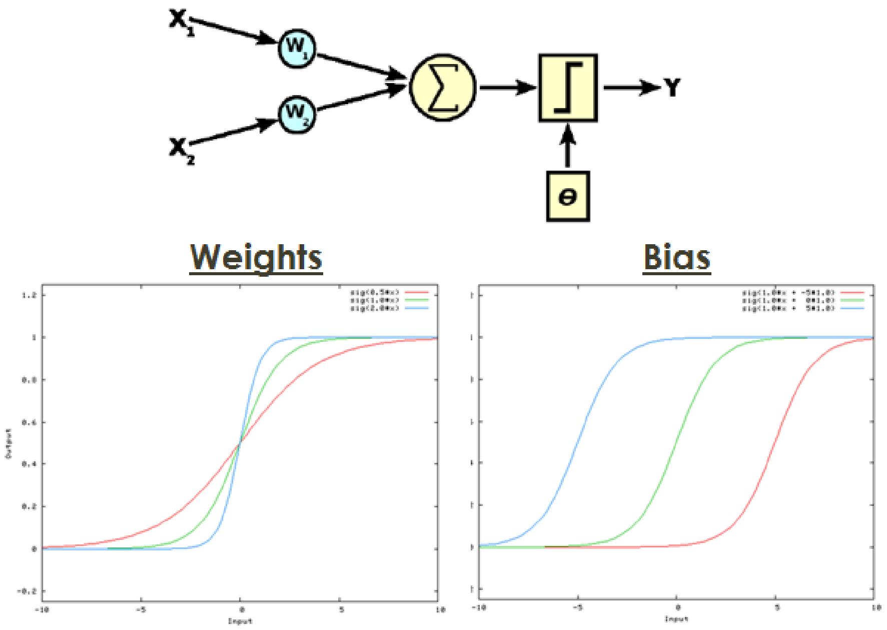


Figure 6.3: Weights and Bias Example in Perceptron

1. The **interconnection pattern** between the different layers. This allows to differentiate two different architectures: *feed-forward networks and feed-back networks*, which are presented in Figure 6.4. The first ones allow the signals to travel one way from input to output, whereas in the second one, the signals travel as loops in the network and the output is connected to the input. In prediction, for example, a forward propagation pattern is used, whereas when calculating the *cost function* the back propagation algorithm is used.

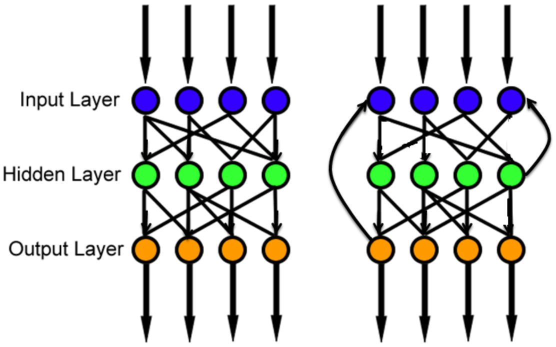


Figure 6.4: Forward and Back propagation Example

2. The **learning process** for updating the weights of the interconnections. A learning rule is defined as a procedure for modifying the weights and biases of a network. The learning rule is applied to train the network to perform some particular task. It determines the specificity of the networks, making them special for different tasks. We can differentiate between *supervised and non supervised learning.* The first one will be used when the classes of the data are known with the purpose of setting the correct weights, that is for the training process. This sorts of problems are Classification problems. The network adjustment is the result of the estimation of the parameters, which is constantly obtained by minimizing a *cost function*. The second one will be used when the classes are not known with the purpose of

discovering which is the correct output class. This is a Clustering problem.

3. The **activation function** that converts a neuron's weighted input to its output activation. This transformation determine the different kind of networks, except from the input nodes.

6.2.1 Problems

According to [3], several problems should be faced during the implementation. In the following points, the most important problems will be discussed.

Under-fitting (High Bias)

It occurs when a statistical model or machine learning algorithm cannot capture the underlying trend of the data. Specifically, under-fitting occurs if the model or algorithm shows low variance but high bias of the estimation. The bias of the estimation is referred to the difference between the gold standard value and the estimated one. Under-fitting is often a result of an excessively simple model.

Over-fitting (High Variance)

It occurs when a statistical model describes random error or noise instead of the underlying relationship. Over-fitting generally occurs when a model is excessively complex, such as having too many parameters relative to the number of observations. A model that has been over-fit will generally have poor predictive performance, as it can exaggerate minor fluctuations in the data. There are two possible solutions to deal with over-fitting while keeping all the features. The first one is called *early stopping* and consists in dividing the data set into three different subsets: training, testing and also a validation set. The validation set will decide when to stop training depending on its error. The second solution is called *regularization* and consists of reducing the magnitude of the theta parameters. When regulated, our model becomes much simpler.

In Figure 6.5, an example can be seen. Imagine that, a simple parameter such as the tumor size is compared with the risk of malignancy, then different responses from our data can be drawn.

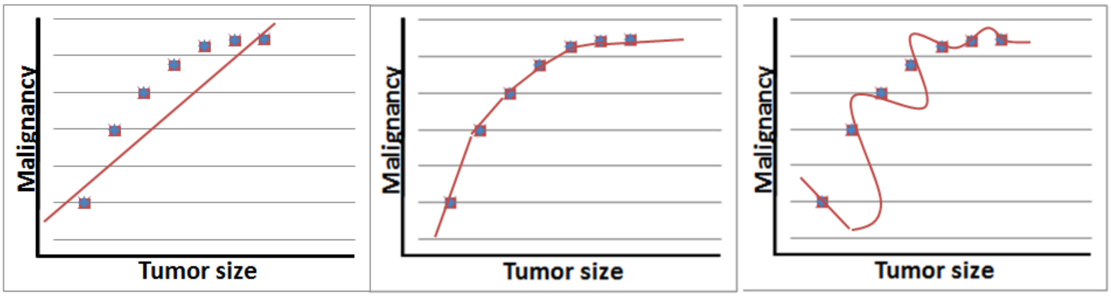


Figure 6.5: Underfitting, Suitable and Overfitting Model Relationship

Besides, in order to check a possible over-fitting problem, the *learning curve* can be calculated. This curve plots the training set error, the testing error and the CV error (in case of the early stopping method) as a function of the number of training images set. Therefore, we will start learning with just one image, then with two and so on. As ten different divisions of the data have been made, different learning curves can be obtained. This differences depend on the random image group which is included for learning in the training state, for example, a difficult or an easy differentiable ultrasound image. An example can be seen in Figure 6.6, which shows three different cases. Depending on the data features and data divisions, different results can be obtained.

In the *first one*, any method against over-fitting has been applied. Therefore, we are dealing with it specially at the beginning, until the training model achieves to make a generalization of the data (some point around 30 images). From that point on, the errors decrease meeting a value around 0.2 error. We make three divisions of the data in order to compare it with the next method. In the *second case,* the early stopping method is applied and the CV set error is used during training in order to decide when to stop the network learning procedure. Therefore, at the beginning a high training error is obtained, because the network has not enough images to classify correctly. This error decreases, when using more images. In the *last case*, regularization is applied. It is enough to make two divisions of the data, so that more images can be used for training. However, at the beginning the over-fitting effect is not removed. It affects less than in the first case, but more data is needed in order to realize it. From now on, the early stopping method will be used in our implementation.

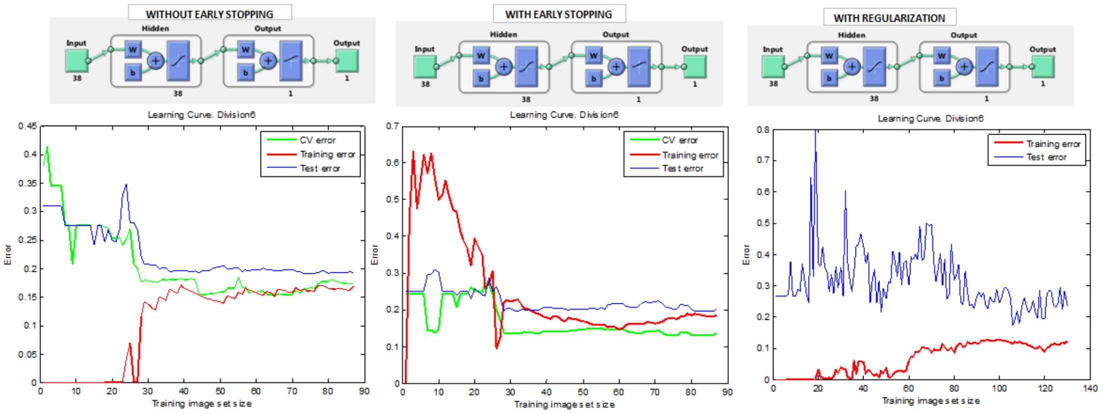


Figure 6.6: Learning Curve for Division 6

6.2.2 Implementation

According to [12], we will proceed to select the data, create and train a network, and evaluate its performance. Pattern recognition networks are feed-forward networks that can be trained to classify inputs according to target classes. Implementing a NN models follows a number of systemic procedures. Regarding [6], there are six basics steps: (1) pre-processing data, (2) building the network, (3) data division, (4) parameters initialization, (5) training, and (6) testing the performance of the model. The diagram implementation steps can also be seen in Figure 6.7.

Data pre-processing

As it has been said before, the data needs to be rescaled before entering the classifier. The reason is to make it easier for the neural network to adapt to the inputs, since mixing variables with large and small magnitudes will confuse the learning algorithm on the importance of each variable. Apart from normalization, the data can also be randomized.

Building the network

In this stage, it is needed to choose the network architecture. The best architecture to use depends on the type of problem to be represented by the network.

The **Activation function** converts a neuron's weighted input to its output activation. MATLAB provides a wide range of built-in transfer functions. In Figure 6.8, four examples can be seen, Linear (purelin), Hyperbolic tangent sigmoid (tansig), Logarithmic sigmoid (logsig) and Softmax

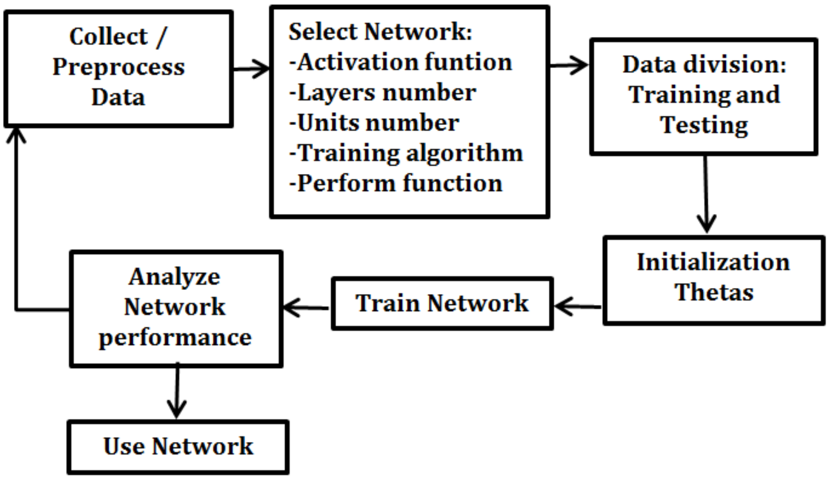


Figure 6.7: Implementation

transfer function. It is important for the activation functions to be differentiable in order to apply certain algorithms. For pattern recognition, it is advisable to use either the sigmoid function or the softmax one. In the hidden units, the sigmoid activation functions are usually preferable. A small change in the weights will usually produce a change in the outputs, which makes it possible to tell whether that change in the weights is good or bad.

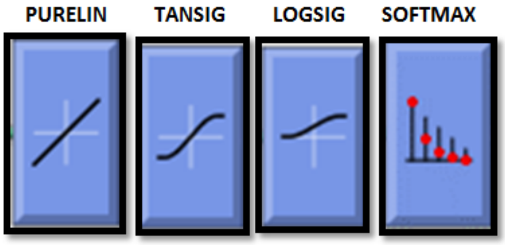


Figure 6.8: Activation function examples

According to the **layers number**, we can differentiate between one input layer, one output layer and a certain number of hidden layers. A good default is to start with one hidden layer and increase the number if needed. However, in practice, it is uncommon to see neural networks with more than two or three hidden layers.

Regarding the **units number**, we clearly see that in the input layer, this number is determined by the number of input variables: significant features in ST or coefficients in PCA. However, the number of units in the last layer depends on the number of classes we have, i.e. benign and malignant groups. In Figure 6.9 and 6.10, two networks will be represented. With green color we will refer to several inputs and with the blue one to particular ones. Inside of each layer the weight and bias boxes will be represented. Therefore, we can use:

- Two output neurons, one for each class, with a softmax activation function. If a1 and a2 are the outputs of the two output neurons, $P(y = y1|x) = a1$ and $P(y = y2|x) = a2$ with $a1 + a2 = 1$.
- A single output neuron with a sigmoid activation function. If a is the output of the neuron,

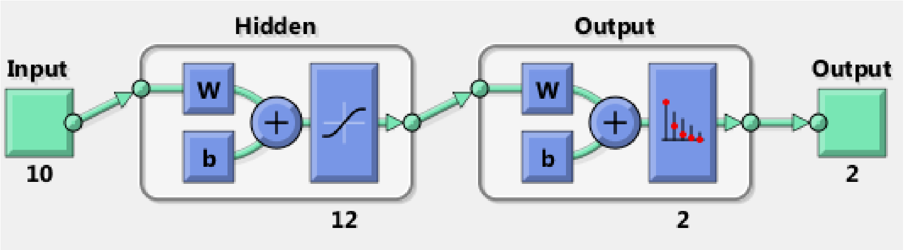


Figure 6.9: Two outputs network

we can set $P(y = y1|x) = a$ and $P(y = y2|x) = 1 - a$.

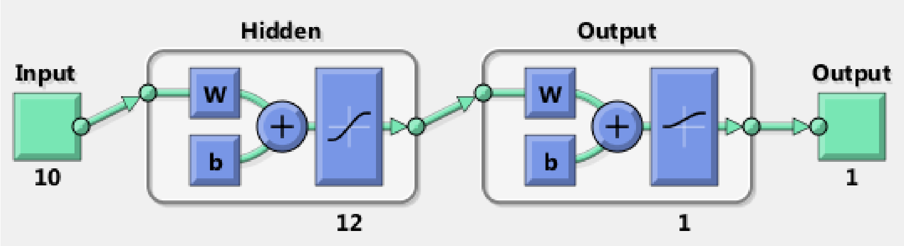


Figure 6.10: Single output network

The first option has twice more parameters in the last layer and thus, has more flexibility and can potentially model more complicated relationships. The second option has twice less parameters, and thus, is less prone to over-fitting. Finally, in the hidden layer, it is advisable that the number of units is greater than the number of features. We will call the different networks by the number of units in its successive layers: *(Input layer - Hidden layer 1 - ... - Hidden layer n-Output layer)*.

The **Training function** is the algorithm that will update the network during the training. We will proceed to evaluate seven different back-propagation training algorithms in order to see, which one is the most appropriate for our data. Some of them require more memory and computation time, others perform better in function fitting (nonlinear regression) than pattern recognition problems. The algorithms include: Resilient Backpropagation (RP), Scaled Conjugate Gradient (SCG), Powell-Beale Conjugate Gradient(CGB), Fletcher-Powell Conjugate Gradient (CGF), Polak-Ribiere Conjugate Gradient (CGP), One Step Secant (OSS) and Variable Learning Rate Gradient Descent (GDX).

The **Perform function** is used to measure a network's usefulness during training. For example, the mean squared error or the cross-entropy function. Minimizing them leads to good classifiers. We will use the MSE.

- **Mean square error (MSE)**: Measure of the differences between the target value from the gold standard and the predicted value by the neural network model. It will take into account the closeness of a prediction to its original value.

$$MSE = \frac{1}{N}\sum(\text{Target} - \text{Output})^2 \tag{6.1}$$

- **Average cross entropy error (ACE)**: Measure of the product of the logarithm of each computed output multiplied by its corresponding target. It leads to faster training.

$$ACE = -\frac{1}{N}\sum(\ln(\text{Output}) * \text{Target}) \tag{6.2}$$

Data division

One of the major advantages of neural nets is their ability to *generalize*. That means that the net could classify data that has never seen before as the learning data. However, as many developers, we have a small part of possible patterns (145 images) so to reach the best generalization, the data set will be split. In the K-fold cross validation section we required two main groups (training and testing). However, as explained before, in order to avoid NN over-fitting, we will use the early stopping procedure. Therefore, the training group will be further divided into two sets. The final division, following this method, will be:

1. The **training set** is used to train a neural net. The 90% of the data is used here, but we can differentiate two subsets:
    - **Training set**: It will help during the training, by adjusting the weights of the network and making the actual output close to the target one. In this way, the error of this data set is minimized during the training stage. It will allocate the 80% of the total data, which is equivalent to around 117-119 patients.
    - **Validation set (CV)** is used to determine the performance of a neural network on patterns that are not trained during learning. It will allocate the 10% of the total data, which is equivalent to around 13-15 patients. It can help to find the best neural network configuration and training parameters, for example by minimizing over-fitting. The weights of the network are not being adjusted with this data set, but it is being verified that any increase in accuracy over the training data set actually yields to an increase in accuracy over a validation set, that has not been shown to the network. If the accuracy over the training data set increases, but the accuracy over the validation data set stays the same or decreases, then the neural network is over-fitting and it should stop training. This validation method is also called, early stopping.
2. The **testing set** for finally checking the performance of a neural net. This set will also allocate the 10%, which means around 13-15 patients. It is collected separately from training and validation sets to help ensure independence. It remains as an unbiased estimate of the network.

Therefore, the CV and Testing set will both have the same size. It is also important to be sure that all those sets contain both positive and negative examples with the purpose to learn to separate ones from the others. Classifiers are trained to accept or detect positive samples, and reject negative ones. As explained in the previous section, we will require to have the same proportion of benign and malignant samples in each of those sets as in the database. That is, if 145 images correspond to the 100%, then 106 benign and 39 malignant images will correspond to the 73% and 27% respectively in each of the sets. In Figure 6.11, a data division example is presented. The training group set of image characteristics will be given with their gold standard value, whereas the test set group will be used to see its performance and check their results.

The partitioning of input data is performed randomly with the certain ratio of input entities mentioned and following the K-fold cross validation division, explained at the beginning. In Figure 6.12, the first three random divisions of the whole data can be seen. Each of the sets has been assigned a color, so that we can easily appreciate its randomness: blue is assigned to training, green to validation and red to the test set.

Parameters Initialization

Before training a neural network, it is also needed to initialize all the parameters. Instead of initializing all the parameters to zero, which works for a Logistic Regression, it is needed to initialize parameters randomly. If thetas are all the same on the first iteration, it can be probed that they would still be equal to each other on the following one, which means that all activation

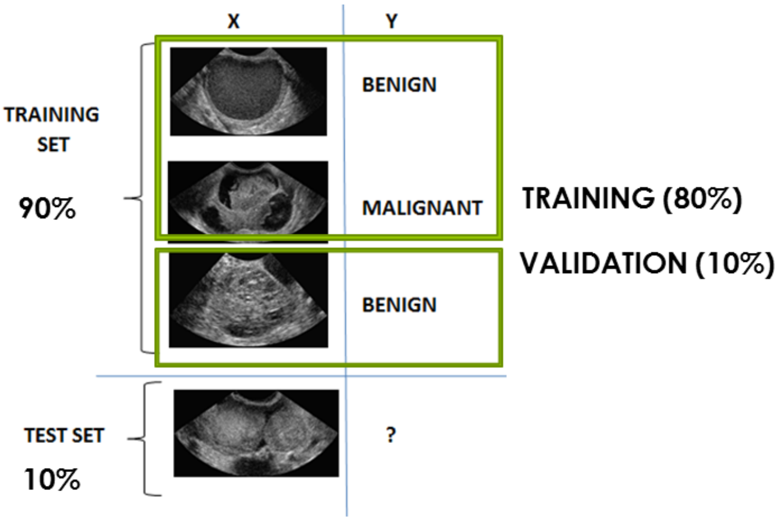


Figure 6.11: Data division

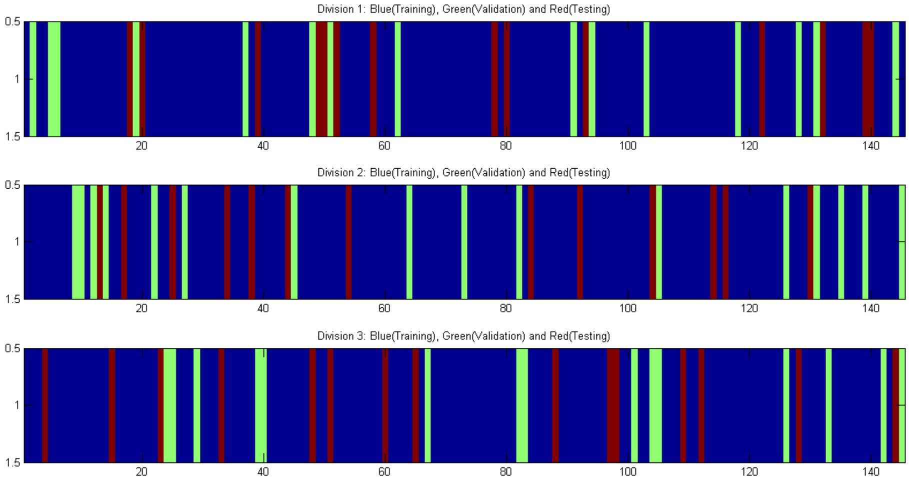


Figure 6.12: Data random divisions

units will have the same values and therefore, they will be redundant. Hence, the results after training the network can vary slightly every time the example is run. However, testing the data for several initial conditions, verify the robustness of the performance. Another option is setting the random seed from the beginning so as to reproduce the same results every time.

Training the network

As we have said before, during the training process the weights are adjusted in order to make the actual outputs close to the target outputs of the network. Besides, the multiple times training (as explained in the CV section) is included. In that way, ten random divisions of the data will be made in order to generalize the results. We can differentiate two options:

- Re-train multiple times and average the outputs of all the testing sets. This will give information about how good is our classifier and it is likely to generalize better to additional new images, because it takes into account all the different subsets.
- Re-train multiple times and take the testing set division with the lowest error, i.e. best performance. The NN group with the highest performance is the best division from our dataset.

Therefore, two testing results will be presented. In this way, it will not be a problem if the easiest images fall into the training group and the difficult ones into the testing group, which may cause a bad result, because the average results are also presented. Besides, we can be sure that at least the test set is divided into 10 disjoint subsets.

Testing the network

The next step is to test the performance of the developed model. At this stage unseen data included in the test set is exposed to the model. In order to evaluate the performance of the developed ANN models quantitatively and verify whether there is any underlying trend in performance of ANN models, statistical analysis involving the mean square error (MSE), classification error (CE), sensitivity, specificity, accuracy and positive predictive value were conducted.

Defining positive as the malignancy group and negative as the benign one, these concepts can be defined as follows. The nomenclature is: True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN).

- **Classification error (CE)**: Measure of the number of false (positive and negative) classifications in the neural network model. It will estimate the effectiveness of the neural network.

$$CE = \frac{\sum \text{FP} + \text{FN}}{\sum \text{Total Population (TP + TN + FP + FN)}} \tag{6.3}$$

- **Sensitivity/ True positive rate (Sn)**: Proportion of malignant images which are correctly identified. It will be presented in percentage.

$$Sn = \frac{\sum \text{TP}}{\sum \text{Condition Positive (TP + FN)}} \tag{6.4}$$

- **Specificity/ True negative rate (Sp)**: Proportion of benign images correctly identified. It will be presented in percentage.

$$Sp = \frac{\sum \text{TN}}{\sum \text{Condition Negative (TN + FP)}} \tag{6.5}$$

- **Accuracy (ACC)**: Measure of how well the binary classifier correctly identifies or excludes a condition, that is the proportion of true results (both benign and malignant) among the total number of cases. It will be presented in percentage.

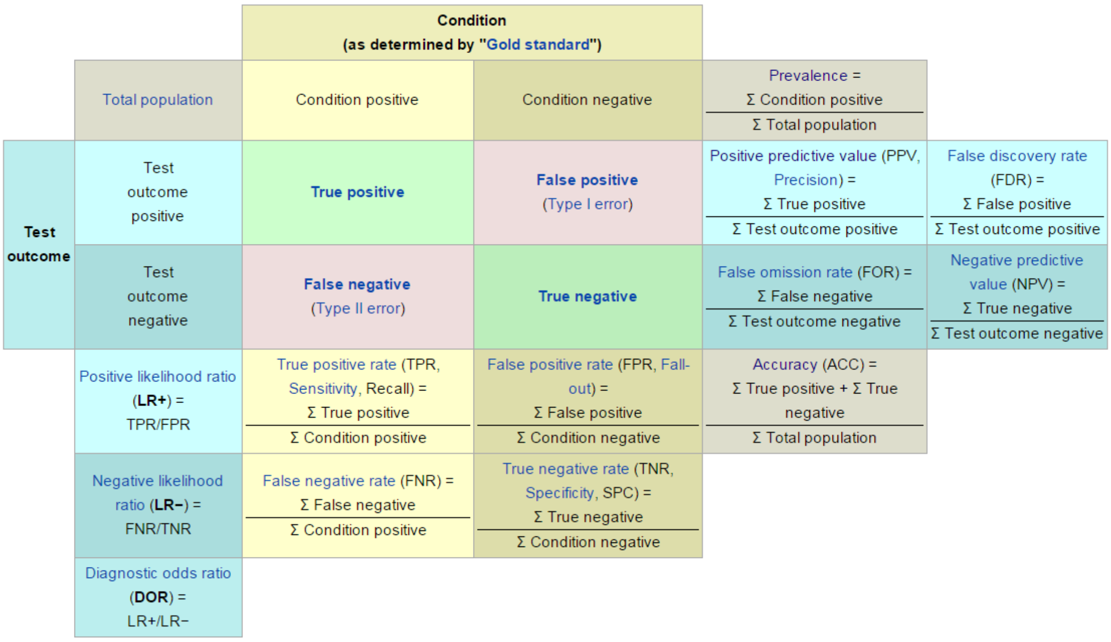
$$ACC = \frac{\sum \text{TP} + \text{TN}}{\sum \text{Total Population (TP + TN + FP + FN)}} \tag{6.6}$$

- **Positive Predictive Values (PPV)**: Proportion of the positive results that are true positive, or in other words, the amount of malignant images that are truly malignant. It will be presented in percentage.

$$PPV = \frac{\sum \text{TP}}{\sum \text{Test outcome Positive (TP+ FP)}} \tag{6.7}$$

A perfect predictor would be described as 100% sensitive (all malignant images are identified as malignant) and 100% specific (all benign are identified as benign). In the same way, an accuracy of 100% means that the measured values are exactly the same as the given values. And a 100% in PPV will show that there are no false benign images. On the opposite side, the classifier will return a NaN, if there is no positive images identified. All these concepts and many others are reflected in Figure 6.13. The confusion matrix can be seen, which allows visualization of the performance of an algorithm.

Besides, it should also be taken into account that the PPV value can only be estimated using data from a cross-sectional study in which valid prevalence estimates may be obtained. In contrast, the sensitivity and specificity can be estimated from case-control studies. In our case, the real proportion in the appearance of benign or malignant adnexal mass is taken into account, so that all parameters can be calculated.



		Condition **(as determined by "Gold standard")**			
	Total population	Condition positive	Condition negative	Prevalence = $\frac{\Sigma\ \text{Condition positive}}{\Sigma\ \text{Total population}}$	
Test outcome	Test outcome positive	**True positive**	**False positive** (Type I error)	Positive predictive value (PPV, Precision) = $\frac{\Sigma\ \text{True positive}}{\Sigma\ \text{Test outcome positive}}$	False discovery rate (FDR) = $\frac{\Sigma\ \text{False positive}}{\Sigma\ \text{Test outcome positive}}$
	Test outcome negative	**False negative** (Type II error)	**True negative**	False omission rate (FOR) = $\frac{\Sigma\ \text{False negative}}{\Sigma\ \text{Test outcome negative}}$	Negative predictive value (NPV) = $\frac{\Sigma\ \text{True negative}}{\Sigma\ \text{Test outcome negative}}$
	Positive likelihood ratio (**LR+**) = TPR/FPR	True positive rate (TPR, Sensitivity, Recall) = $\frac{\Sigma\ \text{True positive}}{\Sigma\ \text{Condition positive}}$	False positive rate (FPR, Fall-out) = $\frac{\Sigma\ \text{False positive}}{\Sigma\ \text{Condition negative}}$	Accuracy (ACC) = $\frac{\Sigma\ \text{True positive} + \Sigma\ \text{True negative}}{\Sigma\ \text{Total population}}$	
	Negative likelihood ratio (**LR−**) = FNR/TNR	False negative rate (FNR) = $\frac{\Sigma\ \text{False negative}}{\Sigma\ \text{Condition positive}}$	True negative rate (TNR, Specificity, SPC) = $\frac{\Sigma\ \text{True negative}}{\Sigma\ \text{Condition negative}}$		
	Diagnostic odds ratio (**DOR**) = LR+/LR−				

Figure 6.13: Confusion Matrix

6.2.3 Network programming

In Figure 6.14, a network diagram can be seen. This diagram explains the data work flow in our neural network. As we have said, the data will be randomly divided and each of the sets will be trained and tested in order to save the corresponding performances.

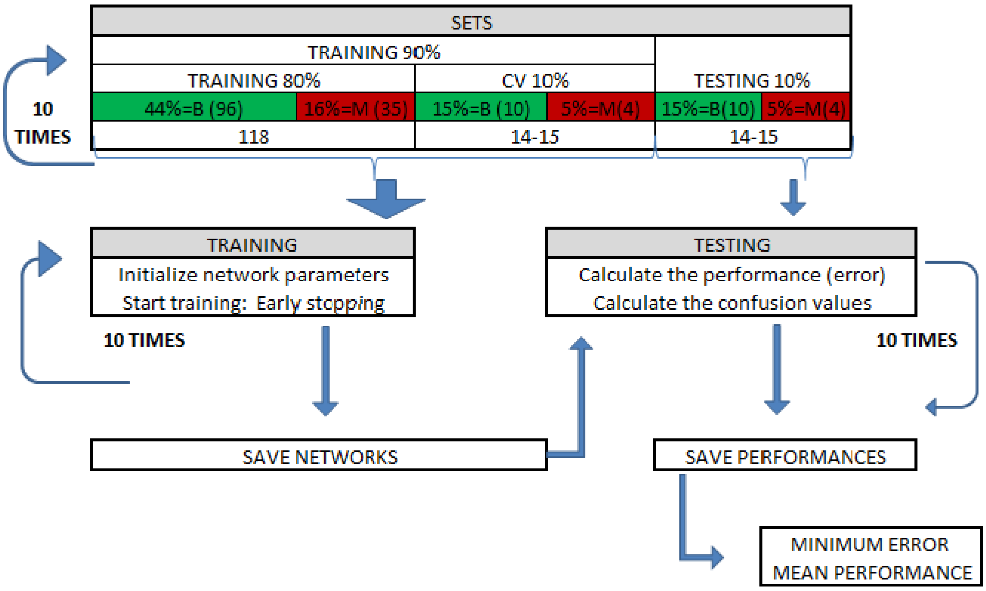


Figure 6.14: Diagram Network

We will proceed to represent different networks configurations and compare their results in order to obtain an optimal NN configuration. We will work on a specific network called, **Multilayer perceptron**, which is a multilayer feed-forward network, that means that it will have three or more layers. We will start with three. The network is trained by the back propagation learning rule.

A commonly used cost function is the mean-squared error, and the goal is to minimize this value between the networks output and the target value over all the examples. This minimization is done by several training algorithms, as we have already mentioned, from where we can emphasize those with *Gradient descent* optimization algorithm. The method calculates the gradient of a loss function with respect to all the weights in the network, as seen in Figure 6.15. However, there are several alternatives of these training algorithms are available.

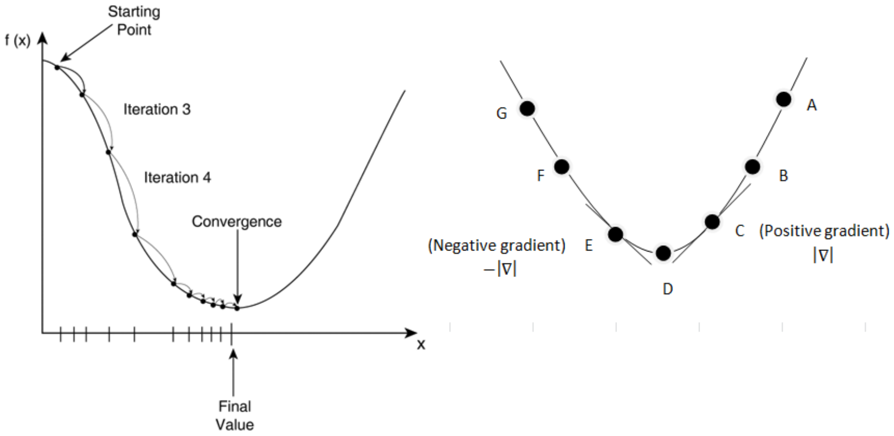


Figure 6.15: Gradient descent

During the training, both the training and CV set will be evaluated. As we have said, this technique is usually called as the *Early stopping*, provided for all the supervised networks. In Figure 6.16, an example can be seen. The first subset is the training set, which is used for computing the gradient and updating the network weights and biases and the second subset is the validation set, whose error is monitored during the training process. The validation error normally decreases during the initial phase of training, as does the training set error. However, when the network begins to over-fit the data, the error on the validation set typically begins to rise. In this moment, if the validation set error increases or keeps constant for a specified number of iterations or usually called epochs, the training is stopped, and the weights and biases at the minimum of the validation error are returned. This network is supposed to be the optimal one, and its parameters will be saved. The test set error is not used during training, but during evaluation of the saved networks. Then the performance error of the independent test set will be calculated, and also the confusion matrix values. Finally, the mean and minimum error results will be presented.

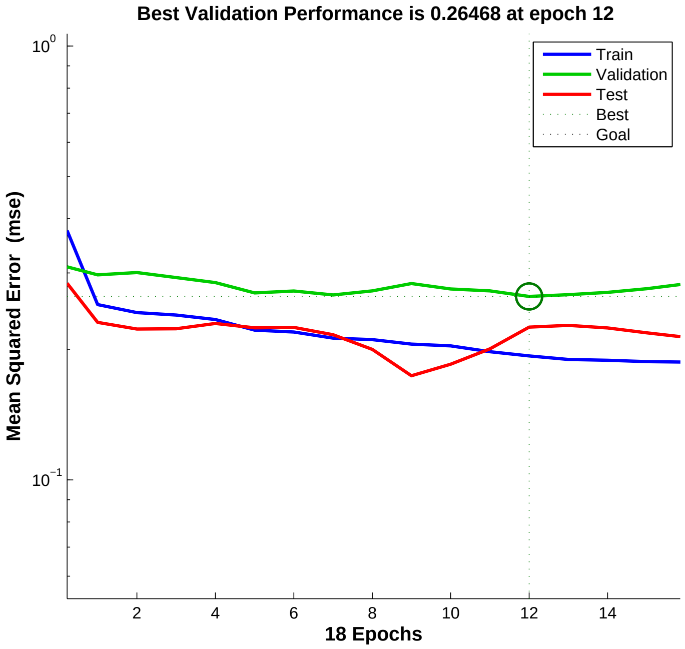


Figure 6.16: Performance (Early stopping)

It is also possible to represent the obtained errors for the different sets. The errors are calculated as the target output, minus the obtained one. Here it is also possible to detect outliers, looking to the specific errors obtained. In Figure 6.17, the blue colors correspond to the training set, the green to the validation and the red ones to the test set. We can see that in this network, the errors are between -0.8 and 0.8.

Algorithm selection

According to the network that corresponds to the ST parameters for the Semi-automated procedure as example, we will proceed to make an study in order to select the proper algorithm. Therefore, we start with 38 input units and 46 units in the hidden layer (20% more). Therefore, this network can be called 38-46-2. We can see, that we do not exceed the 87 units (restriction for the training number of patients). Using small NN would be prone to high bias and under-fitting, as we have few parameters. However, these networks are computationally cheaper. Using a large NN with more parameters are more prone to over-fitting. But in general, large networks usually do better job than smaller networks, although they are more computationally expensive too.

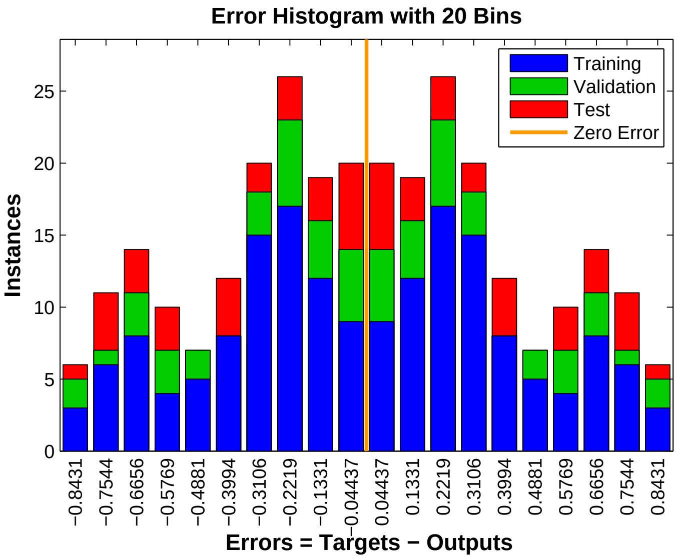


Figure 6.17: Error histogram

To choose the optimal algorithm, in Table 6.1 the results of seven different ones are presented. Mean Squared Error, Classification Error, Sensitivity, Specificity, Accuracy and Positive Predictive Value are included. The average result of the ten data divisions is taken too.

Algorithm	**Network**	**MSE**	**CE**	**Sn**	**Sp**	**ACC**	**PPV**
RP	38-46-2	0,262	0,273	11,66	95,36	72,65	11,66
SCG	38-46-2	0,243	0,303	17,50	88,54	69,63	17,50
CGB	38-46-2	0,241	0,266	15,83	94,36	73,31	15,83
CGF	38-46-2	0,245	0,303	14,16	90,27	69,65	14,16
CGP	38-46-2	0,239	0,260	23,33	92,27	73,93	23,33
OSS	38-46-2	0,250	0,303	18,33	88,45	69,69	18,33
GDX	38-46-2	0,242	0,307	13,33	89,90	69,22	13,33

Table 6.1: Results for the network $38-46-2$

The algorithm with the lowest mean squared error is CGP (Polak-Ribiere Conjugate Gradient) with an MSE error of 0,239 and with a classification error of 0,260. From now on, we will use this algorithm to calculate the following results.

6.2.4 Results

We will present the results of two networks with different activation functions: softmax and sigmoid. The number of units in the hidden layer will be the number of input units plus the 20% and the training algorithm is called Polak-Ribiere Conjugate Gradient. The maximum number of units in the hidden layer will be set to 117, which corresponds to the number of the training set patients. We will present the best division and the average results, which will be different depending on the K-fold cross-validation division of each execution. The best division results will also depend on the size of that test, which will vary between 13 and 15 images.

We will present both the MSE and CE error. The first one will indicate the closeness of a prediction to its gold standard value (Target - Output) and the second one will estimate the effectiveness of the neural network, or in other words how many false (positive and negative) classifications the network has. For the best division network, we will choose the division with the lowest CE. In case that several networks with the same CE are presented, the one with a smaller difference between specificity and sensitivity will be taken.

Besides, as it has been said before, sensitivity, specificity, accuracy and positive predictive value are evaluated. In the best division results, the percentage of right answers in the malignant and benign groups, respectively, is taken into account.

Semi-automated Procedure

In Table 6.2, 6.3 and 6.4, the results for the softmax activation function can be seen. In Table 6.5, 6.6 and 6.7, the results for the sigmoid activation function can be seen. Comparing them, it can be said that:

- Results for ST Semi-automated: Using 38 features leads to better average division results. Sigmoid function (0,284) presents lower CE results than the Softmax (0,328), which means that the values have high accuracy. The MSE error is also low and therefore the predictions are closer to their target values.
- Results for PCA Semi-automated: The 90% and 99% variability present the same CE for the best division in the softmax function. However the 90% has a lower MSE and the Sn and Sp are more balanced. Softmax function for 90% (0,289) presents a lower CE error than sigmoid (0,335).
- Results for Hybrid Semi-automated: In the softmax function, both best divisions present the same results, although the first one has a lower MSE. The average division for 48 features (38 ST and 10 PCA) present the best results in the softmax function (0,262) in comparison with the sigmoid (0,297).

ROI Procedure

In Table 6.8, 6.9 and 6.10, the results for the softmax activation function can be seen. In Table 6.11, 6.12 and 6.13, the results for the sigmoid activation function can be seen. Comparing them, it can be said that:

- Results for ST ROI : Softmax presents the better average results with a lower CE (0,239) in comparison with sigmoid CE (0,249). Therefore the network with 59 features is more appropiate. The best division results have the lowest CE (0,076) with a 100% of Sn and a 90% of Sp.
- Results for PCA ROI : Softmax presents the second best average results with a CE (0,255) in 95% variability in comparison with sigmoid, which has a CE (0,304) in 90% variability. Increasing in variability above 95% does not contribute much.
- Results for Hybrid ROI: Softmax presents better results. The best division presents the same CE error, although the network with 96 features(37 PCA and 59 ST) present the lowest MSE. The average results are also better.

Best division Softmax

Network	**MSE**	**CE**	**Sn**	**Sp**	**ACC**	**PPV**	**%**
41-49-2	0,126	0,142	50,00	100,00	85,71	100,00	2/4,10/10
553-117-2	0,219	0,214	25,00	100,00	78,57	100,00	1/4,10/10

Average division Softmax

Network	**MSE**	**CE**	**Sn**	**Sp**	**ACC**	**PPV**
41-49-2	0,280	0,328	10,83	88,00	67,13	33,57
553-117-2	0,339	0,364	15,83	81,45	63,58	19,47

Table 6.2: Results for ST Semi-automated Softmax

Best division Softmax

Variability	**Network**	**MSE**	**CE**	**Sn**	**Sp**	**ACC**	**PPV**	**%**
90%	7-8-2	0,154	0,133	50,00	100,00	86,66	100,00	2/4,11/11
95%	16-19-2	0,221	0,200	50,00	90,90	80,00	66,66	2/4,10/11
99%	70-84-2	0,129	0,133	50,00	100,00	86,66	100,00	2/4,11/11
100%	144-117-2	0,185	0,200	25,00	100,00	80,00	100,00	1/4,11/11

Average division Softmax

Variability	**Network**	**MSE**	**CE**	**Sn**	**Sp**	**ACC**	**PPV**
90%	7-8-2	0,216	0,289	20,83	89,451	71,02	38,33
95%	16-19-2	0,246	0,316	20,00	86,00	68,31	36,22
99%	70-84-2	0,261	0,289	7,50	94,36	71,02	23,33
100%	144-117-2	0,277	0,322	7,50	89,72	67,73	12,00

Table 6.3: Results for PCA Semi-automated Softmax

Best division Softmax

Combination	**Network**	**MSE**	**CE**	**Sn**	**Sp**	**ACC**	**PPV**	**%**
90% + pv(0,05)	48-58-2	0,132	0,133	50,00	100,00	86,66	100,00	2/4,11/11
95% + pv(0,05)	57-68-2	0,147	0,133	50,00	100,00	86,66	100,00	2/4,11/11

Average division Softmax

Combination	**Network**	**MSE**	**CE**	**Sn**	**Sp**	**ACC**	**PPV**
90% + pv(0,05)	48-58-2	0,240	0,262	22,50	92,27	73,73	58,33
95% + pv(0,05)	57-68-2	0,264	0,890	18,33	90,54	71,06	38,88

Table 6.4: Results for Hybrid Semi-automated Softmax

Best division Sigmoid

Network	MSE	CE	Sn	Sp	ACC	PPV	%
41-49-1	0,147	0,200	50,00	90,90	80,00	66,66	2/4,10/11
553-117-1	0,051	0,066	100,00	90,90	93,33	80,00	4/4,10/11

Average division Sigmoid

Network	MSE	CE	Sn	Sp	ACC	PPV
41-49-1	0,203	0,284	17,50	91,27	71,58	33,33
553-117-1	0,200	0,306	25,83	85,54	69,35	31,11

Table 6.5: Results for ST Semi-automated Sigmoid

Best division Sigmoid

Variability	Network	MSE	CE	Sn	Sp	ACC	PPV	%
90%	7-8-1	0,152	0,230	0,00	100,00	76,92	NaN	0/1,10/10
95%	16-19-1	0,151	0,153	33,33	100,00	84,61	100,00	1/3,10/10
99%	70-84-1	0,212	0,266	25,00	90,90	73,33	50,00	1/4,10/11
100%	144-117-1	0,150	0,200	25,00	100,00	80,00	100,00	1/4,11/11

Average division Sigmoid

Variability	Network	MSE	CE	Sn	Sp	ACC	PPV
90%	7-8-1	0,227	0,335	22,50	82,27	66,45	32,61
95%	16-19-1	0,249	0,303	26,66	85,63	69,69	46,11
99%	70-84-1	0,274	0,400	25,00	72,63	59,90	31,04
100%	144-117-1	0,258	0,317	5,00	91,45	68,26	21,42

Table 6.6: Results for PCA Semi-automated Sigmoid

Best division Sigmoid

Combination	Network	MSE	CE	Sn	Sp	ACC	PPV	%
90% + pv(0,05)	48-58-1	0,195	0,214	25,00	100,00	78,57	100,00	1/4,10/10
95% + pv(0,05)	57-68-1	0,181	0,200	75,00	81,81	80,00	60,00	3/4,9/11

Average division Sigmoid

Combination	Network	MSE	CE	Sn	Sp	ACC	PPV
90% + pv(0,05)	48-58-1	0,204	0,309	2,50	93,45	69,01	14,28
95% + pv(0,05)	57-68-1	0,253	0,297	23,33	87,36	70,21	39,79

Table 6.7: Results for Hybrid Semi-automated Sigmoid

Best division Softmax

Network	MSE	CE	Sn	Sp	ACC	PPV	%
58-70-2	0,117	0,076	100,00	90,00	92,30	75,00	3/3,9/10
570-117-2	0,131	0,133	75,00	90,90	86,66	75,00	3/4,10/11

Average division Softmax

Network	MSE	CE	Sn	Sp	ACC	PPV
58-70-2	0,236	0,239	27,50	94,45	76,04	61,45
570-117-2	0,280	0,300	12,50	91,09	70,00	28,33

Table 6.8: Results for ST ROI Softmax

Best division Softmax

Variability	Network	MSE	CE	Sn	Sp	ACC	PPV	%
90%	11-13-2	0,162	0,200	75,00	81,81	80,00	60,00	3/4,9/11
95%	27-32-2	0,161	0,153	33,33	100,00	84,61	100,00	1/3,10/10
99%	83-100-2	0,225	0,230	33,33	90,00	76,92	50,00	1/3,9/10
100%	144-117-2	0,230	0,230	0,00	100,00	76,92	NaN	0/3,10/10

Average division Softmax

Variability	Network	MSE	CE	Sn	Sp	ACC	PPV
90%	11-13-2	0,249	0,347	20,00	81,81	65,28	22,52
95%	27-32-2	0,248	0,274	15,83	93,54	72,55	54,66
99%	83-100-2	0,286	0,290	5,83	95,00	70,97	35,00
100%	144-117-2	0,325	0,368	2,50	85,45	63,12	2,85

Table 6.9: Results for PCA ROI Softmax

Best division Softmax

Combination	Network	MSE	CE	Sn	Sp	ACC	PPV	%
90% + pv(0,05)	69-83-2	0,154	0,200	50,00	90,90	80,00	66,66	2/4, 10/11
95% + pv(0,05)	95-114-2	0,143	0,200	75,00	81,81	80,00	60,00	3/4, 9/11

Average division Softmax

Combination	Network	MSE	CE	Sn	Sp	ACC	PPV
90% + pv(0,05)	69-83-2	0,262	0,283	15,83	92,27	71,63	33,33
95% + pv(0,05)	95-114-2	0,246	0,261	24,16	92,54	73,78	51,66

Table 6.10: Results for Hybrid ROI Softmax

Best division Sigmoid

Network	**MSE**	**CE**	**Sn**	**Sp**	**ACC**	**PPV**	**%**
58-70-1	0,068	0,066	100,00	90,90	93,33	80,00	4/4,10/11
570-117-1	0,092	0,066	75,00	100,00	93,33	100,00	3/4,11/11

Average division Sigmoid

Network	**MSE**	**CE**	**Sn**	**Sp**	**ACC**	**PPV**
58-70-1	0,244	0,304	30,83	83,90	69,59	30,92
570-117-1	0,183	0,249	49,16	84,81	75,04	54,16

Table 6.11: Results for ST ROI Sigmoid

Best division Sigmoid

Variability	**Network**	**MSE**	**CE**	**Sn**	**Sp**	**ACC**	**PPV**	**%**
90%	11-13-1	0,107	0,133	50,00	100,00	86,66	100,00	2/4,11/11
95%	27-32-1	0,228	0,316	66,66	80,00	76,92	50,00	2/3,8/10
99%	83-100-1	0,206	0,266	50,00	81,81	73,33	50,00	2/4,9/11
100%	144-117-1	0,254	0,266	0,00	100,00	73,33	NaN	0/4,11/11

Average division Sigmoid

Variability	**Network**	**MSE**	**CE**	**Sn**	**Sp**	**ACC**	**PPV**
90%	11-13-1	0,208	0,304	36,66	81,72	69,54	46,83
95%	27-32-1	0,228	0,316	34,16	80,90	68,31	35,18
99%	83-100-1	0,293	0,399	25,00	72,90	60,90	20,66
100%	144-117-1	0,289	0,324	5,83	90,63	67,58	15,00

Table 6.12: Results for PCA ROI Sigmoid

Best division Sigmoid

Combination	**Network**	**MSE**	**CE**	**Sn**	**Sp**	**ACC**	**PPV**	**%**
90% + pv(0,05)	69-83-1	0,123	0,142	50,00	100,00	85,71	100,00	2/4,10/10
95% + pv(0,05)	95-114-1	0,195	0,282	75,00	90,00	71,78	47,50	3/4,9/10

Average division Sigmoid

Combination	**Network**	**MSE**	**CE**	**Sn**	**Sp**	**ACC**	**PPV**
90% + pv(0,05)	69-83-1	0,218	0,287	28,33	86,90	71,22	49,44
95% + pv(0,05)	95-114-1	0,195	0,282	27,50	87,90	71,78	47,50

Table 6.13: Results for Hybrid ROI Sigmoid

6.2.5 Summary

This higher specificity presented in all the average cases, may be due to the fact that there are more negative (10-11 benign) than positive (3-4 malignant) samples in the test group. Therefore, the network is more specialized in recognizing negatives than positives. Since the probability of a sample being negative is much higher than the probability of being positive, the network achieves a greater number of total correct classified samples by classifying most of them as negatives. A possible solution, if we want to achieve more balanced results, would be either to adopt a down-sampling strategy like in [4] to form a regular dataset for classification or to insert a penalty whenever a positive sample is classified incorrectly. However, in this way, we will be losing the benign and malignant probability of appearance in the population. More images in the database will be useful to overcome this problem.

We will mainly focus on the average results, because although they are different on each code execution due to the fact of the different K-fold cross validation, the standard deviation of all of them is not high. The more images we get, the less standard deviation we will have.

In Figure 6.18, different PCA variabilities will be compared in order to see how they affect to the average results. We can clearly see how the more PCA variability is explained, the more negative samples are identified. However, the positive samples percentage decrease. As we have said before, this can be due to the imbalanced dataset. We have also enclosed in blue the best variability results for each case, which is often 90% or 95%.

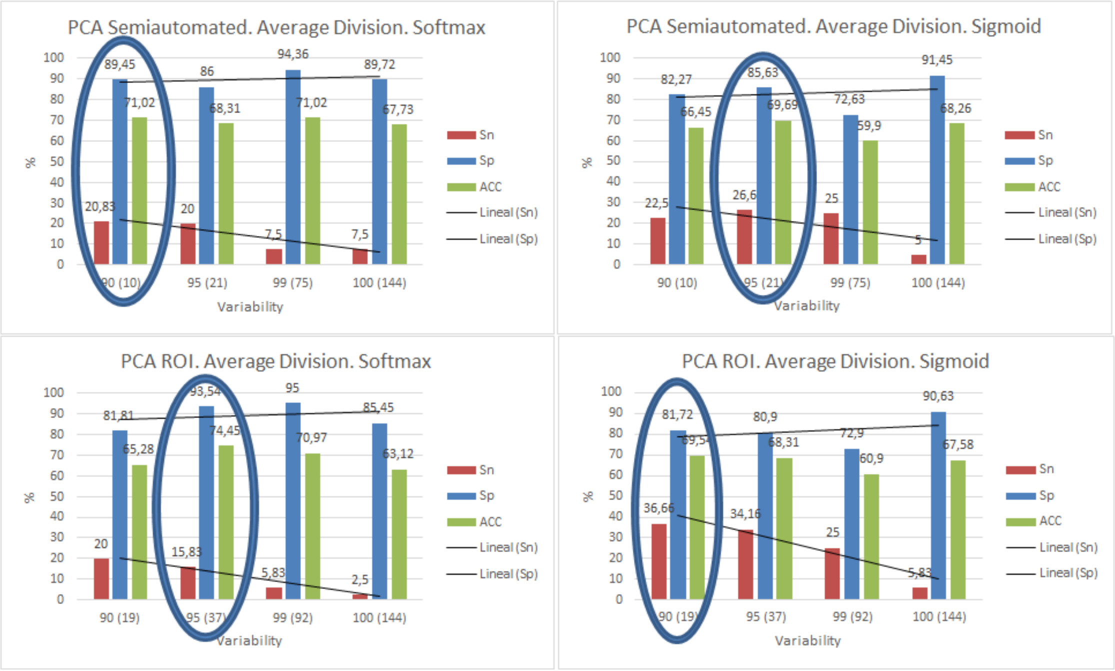


Figure 6.18: PCA variabilities comparison

If the enclosed PCA variabilities are chosen, a simple comparison with the other methods can be done. In Figure 6.19, it can be seen that ST is the one, which provides the best results in most of the cases. ROI average results overcome the SA results specially when dealing with the positive samples detection.

In Table 6.14 and 6.15 , the best division and average results will be selected for each of the three dimensional reduction methods (DRM) and activation functions (AF). The ROI results

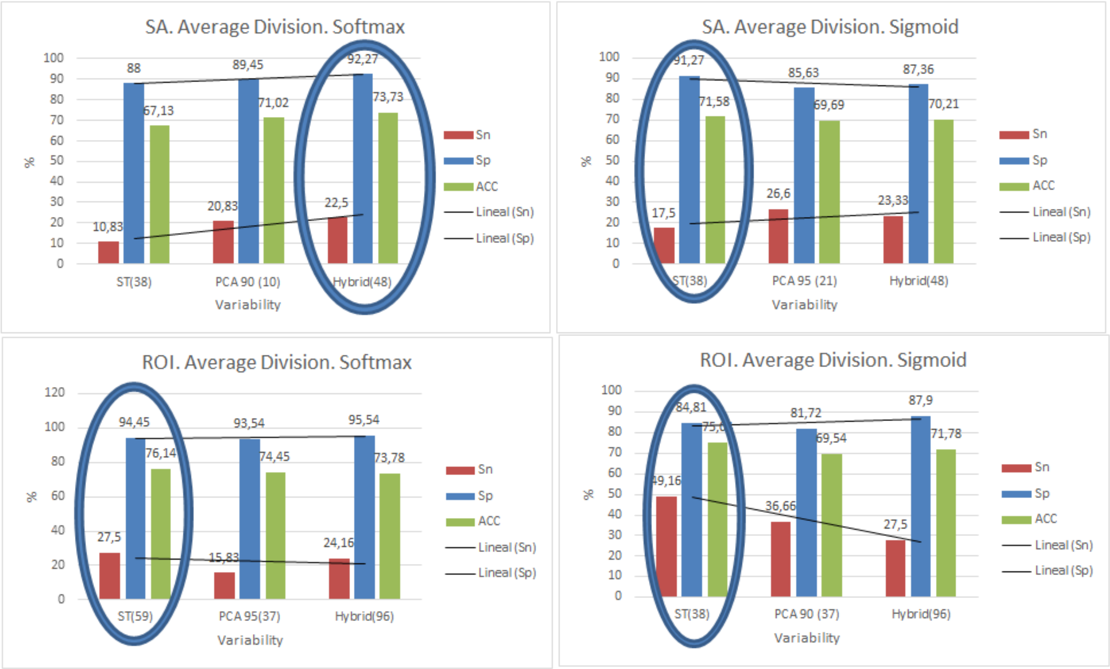


Figure 6.19: All methods comparison

present a lower CE (0,239) in comparison with SA (0,262). This means that they can identify around 76,1% and 73,8% of the results correctly. In the best division results, both SA and ROI present the same CE (0,066), which means the 93,4% of the images in the test set. However, the MSE error is a bit smaller in the SA method.

Average division								
DRM	**AF**	**Network**	**MSE**	**CE**	**Sn**	**Sp**	**ACC**	**PPV**
SA Hybrid	Softmax	48-58-2	0,240	0,262	22,50	92,27	73,73	58,33
ROI ST	Softmax	58-70-2	0,236	0,239	27,50	94,45	76,04	61,45

Table 6.14: Summary average results for SA and ROI

Best division									
DRM	**AF**	**Network**	**MSE**	**CE**	**Sn**	**Sp**	**ACC**	**PPV**	**%**
SA ST	Sigmoid	553-117-1	0,051	0,066	100,00	90,90	93,33	80,00	4/4,10/11
ROI ST	Sigmoid	58-86-1	0,068	0,066	100,00	90,90	93,33	81,00	4/4,10/11

Table 6.15: Summary best division results for SA and ROI

From these results, we can conclude that there is a big difference between the best division and the average results. This can be due to the fact that the images are quite heterogeneous. ST and Hybrid method seem to be the more useful dimensional reduction methods. According to the activation function, the softmax function presents the best average results and the sigmoid one the best division ones.

Chapter 7

Conclusions

Throughout this project a CAD technique for discriminating benign from malignant ultrasound images has been developed. With this aim a feature extraction process, a dimensionality reduction step and a neural network approach have been implemented. It is formulated in MATLAB.

I want to underline that one of the main contributions in this work is the importance of working with just one optimal image from each of the patient volumes in order to create a generalizable classifier. Besides, the database has been created following a heterogeneous mix of pathologies and taking into account the probability of appearance in the population of benign and malignant adnexal masses. Although the presented features have been previously described in different articles, the present work is the first one to provide such a wide collection gathering several of them in one project, with the aim to provide more information to the classifier. These seven features represent a total of 591 different values for each image. Therefore, more importance to the dimensionality reduction step has been also given.

In order to deal with the optimal ultrasound image of the volumes database, two procedures have been tested: a semi-automated and a ROI one. The ultrasound images have been manually segmented when obtaining the ROI, however, manual tracing is an available function in ultrasound machines so that bias selection should not be a significant disadvantage. One goal was to analyze the results for both methods. It can be concluded from the dimensionality reduction process, that with the ROI procedure not only more significant features are achieved in the Student test but also more PCA coefficients for the same percentage of variability than the Semi-automated method. Therefore, it can be assured that with the ROI image more specific information is obtained.

Having the optimal features selected, a neural network for pattern recognition has been implemented. Different configurations have been made in order to achieve a good combination for both, true positives and true negatives rate. Regarding to the problems associated with NN, we have tried to overcome them by using different methods: a complicate model against under-fitting and the early stopping method against over-fitting. The ROI procedure presents the best average results with a classification error of 0,239 (76,04% correctly identifications) with a sensitivity of $27,50\%$ and a specificity of $94,45\%$. The best division results are presented in both SA and ROI with a classification error of 0,066 (93,4% correct identifications) with a sensitivity of $100,00\%$ and a specificity of $90,90\%$.

As future work, more images can be included in order to continue testing this network. A different image classification technique can also be used in order to select just the most uncertain images for the gynecologists and to be more specific in a certain image group. Different configurations of this network or any other classification algorithm can be made with the aim to improve the results too.

Bibliography

[1] 4D View. Users manual. *GE Healthcare.* 8

[2] Manual of diagnostic ultrasound, Volume 2. World Health Organization. *Gynaecology. Uterus and Ovaries.*

[3] Associate Professor at Stanford University. Andrew NG. Machine learning. 45

[4] Khazendar et al. Automated Classification of Static Ultrasound Images of Ovarian Tumours Based on Decision Level Fusion. *6th Computer Science and Electronic Engineering Conference (CEEC). University of Essex, UK*, 2014. 12, 61

[5] Manoj Kumar Biswas et al. Fractal dimension estimation for texture images: A parallel approach. *Pattern Recognition Letters 19 309313*, 1998. 16

[6] T. Hagan Martin et al. Neural network design. 2nd edition. 46

[7] U Rajendra Acharya et al. Evolutionary Algorithm-Based Classifier Parameter Tuning for Automatic Ovarian Cancer Tissue Characterization and Classification. *Ultraschall UNDI UiM-1054 20.11.12 Reemers Publishing Services GmbH*, 2012. 18, 19, 31

[8] U Rajendra Acharya et al. Ovarian Tumor Characterization and Classification: A class of GyneScanTM Systems. *34th Annual International Conference of the IEEE EMBS San Diego, California USA*, 2012. 16, 21, 31

[9] U Rajendra Acharya et al. Ovarian Tumor Characterization and Classification Using Ultrasound: A New Online Paradigm. *Society for Imaging Informatics in Medicine 2012*, 2012. 21, 31

[10] U Rajendra Acharya et al. Ovarian Tumor Characterization using 3D Ultrasound. *Technology in Cancer Research and Treatment. ISSN 1533-0346*, 2012. 13, 22, 31

[11] J.-K. Kämäräinen J. Ilonen and H. Kälviäinen. Efficient Computation Of Gabor Features. 24

[12] Martin T. Hagan Mark Hudson Beale and Howard B. Demuth. Neural Network Toolbox Matlab. Users Guide R2014a. 46

[13] Fei PENG Min LONG. A Box-Counting Method with Adaptable Box Height for Measuring the Fractal Feature of Images. *Radioengineering, VOL. 22, NO. 1, APRIL 2013.* viivii, 17

[14] Campbell K. Mishell DR Jr Grimes DA 3. Koonings PP. Relative frequency of primary ovarian neoplasms: A 10-year review. *Obstetrics and Gynaecology. 1989;74:9216.* 1

[15] Richard E. Woods Rafael C. Gonzalez. Digital image processing.

[16] Shapiro and Stockman. Computer Vision: Mar 2000. Texture.

[17] Zachary R. Smith and Craig S. Wells. Central Limit Theorem and Sample Size. University of Massachusetts Amherst. 31

[18] The Johns Hopkins University. Pelvic Ultrasound. viivii, 2