People's Democratic Republic of Algeria
Ministry of Higher Education and Scientific Research

IBN KHALDOUN UNIVERSITY OF TIARET

Dissertation

Presented to:
FACULTY OF MATHEMATICS AND COMPUTER SCIENCE
DEPARTEMENT OF COMPUTER SCIENCE

in order to obtain the degree of

MASTER

Specialty: software engineering

Presented by:

LABDI Mohamed Alaa Eddine

On the subject

Comparative study of dimensionality reduction techniques in mammographic images

Defended publicly on 07/2023 in Tiaret in front the jury composed of:

Mr CHIKHAOUI Ahmed	MCA	Tiaret University	Chairman
Mr CHENINE AEK	MAA	Tiaret University	Supervisor
Mr DAOUD Mohamed Amine	MAA	Tiaret University	Examiner

2022-2023

DEDICATION

To my dear parents, whose love, patience, unwavering support and encouragement have been instrumental in my academic journey. Their belief in my abilities and faith in my potential have given me the strength and motivated me to overcome challenges and persevere. I am grateful for their guidance and wisdom.

To my siblings, thank you for being my pillars of support, my confidants, and my source of inspiration. Your love, understanding and belief in me have made this endeavor more meaningful and fulfilling.

To all my friends, thank you for being there through the highs and lows, for celebrating the successes and providing solace during the challenges. Your presence has made this experience truly memorable.

I dedicate this achievement to each one of you, as a reflection of our shared values, support, and the bond we hold dear. Thank you for always being there for me.

Acknowledgements

First and foremost, we extend our deepest gratitude to the Almighty God for granting us the strength, guidance, and patience to successfully complete this modest work. Without His blessings, this endeavor would not have been possible.

We would like to take this opportunity to express our heartfelt appreciation to all those who have contributed to the successful completion of this project. we are sincerely grateful for your involvement.

Firstly, I would like to extend my heartfelt gratitude to my dedicated and supportive supervisor, **Dr CHENINE AEK**. Your guidance, patience, and invaluable insights have led me to the successful completion of this research project. I am truly grateful for the countless hours you have devoted to providing valuable feedback, sharing your expertise, and pushing me to reach my full potential.

I would like to extend my sincere appreciation to the esteemed members of the jury, **Mr. CHIKHAOUI Ahmed** and **Mr. DAOUD Mohamed Amine**. Your expertise, critical feedback, constructive suggestions, thorough review, astute observations, and thoughtful comments have added significant value to the scholarly contribution of this research.

I am privileged to have had the opportunity to present my work before such esteemed individuals who are renowned in their respective fields. Your presence and active engagement during the defense have been truly encouraging and have motivated me to strive for excellence.

I would like to express my deep gratitude to every individual who has contributed to the success of this project.

Your hard work, dedication, and creativity have made this project a reality. I am so grateful for your contributions, and I am proud to have worked with such a talented and dedicated team.

Abstract

Breast cancer is a leading cause of cancer death among women. Early detection and diagnosis are essential for improving survival rates. In this study, we developed a computer-aided diagnosis (CAD) system based on machine learning approach for breast cancer detection. The proposed system is presented to investigate effects of dimension reduction techniques for classifying mammograms. It consists of preprocessing, feature extraction, feature selection, dimension reduction and classification steps. The Region of Interest (ROI) is extracted, and textural features are obtained using Local Binary Patterns (LBP). SelectKBest is employed as a feature selection technique to select the relevant features while Dimensionality reduction techniques, including Principal Component Analysis (PCA), Random Projection (RP), and Linear Discriminant Analysis (LDA), are applied to reduce the dimensionality of the images. K-Nearest Neighbors (KNN) and Support Vector Machines (SVM) with Sigmoid, Polynomial and Radial Basis Function kernels are used. The objective is to compare the performance of dimensionality reduction techniques and feature selection using these two classifiers. The initial results show that using classifiers alone did not achieve high accuracy. However, combining SelectKBest with PCA, RP, or LDA resulted in significant accuracy improvements. The best accuracy of 100% was achieved when combining SelectKBest with specific kernels of SVM and dimension reduction techniques. In conclusion, this research demonstrates the effectiveness of dimensionality reduction techniques, especially when combined with a feature selection technique which is SelectKBest in our case, in improving the classification accuracy for the detection of abnormalities in mammograms.

Keywords: mammographic images, machine learning, feature extraction, feature selection, dimension reduction, classification, LBP, SelectKBest, PCA, RP, SRP, GRP, LDA, KNN, SVM.

Résumé

Le cancer du sein est l'une des principales causes de décès par cancer chez les femmes. La détection précoce et le diagnostic sont essentiels pour améliorer les taux de survie. Dans cette étude, nous avons développé un système d'aide au diagnostic assisté par ordinateur (CAD) basé sur une approche d'apprentissage automatique pour la détection du cancer du sein. Le système proposé est présenté pour étudier les effets des techniques de réduction de dimension sur la classification des mammogrammes. Il comprend des étapes de prétraitement, d'extraction de caractéristiques, de sélection de caractéristiques, de réduction de dimension et de classification. La région d'intérêt (ROI) est extraite et des caractéristiques texturales sont obtenues à l'aide des motifs binaires locaux (LBP). SelectKBest est utilisé comme technique de sélection de caractéristiques pour choisir les caractéristiques pertinentes, tandis que des techniques de réduction de la dimensionnalité, telles que l'analyse en composantes principales (PCA), la projection aléatoire (RP) et l'analyse discriminante linéaire (LDA), sont appliquées pour réduire la dimensionnalité des images. Les classificateurs des k plus proches voisins (KNN) et les machines à vecteurs de support (SVM) avec des noyaux sigmoïde, polynomial et à fonction de base radiale sont utilisés. L'objectif est de comparer les performances des techniques de réduction de dimension et de sélection de caractéristiques en utilisant ces deux classificateurs. Les résultats initiaux montrent que l'utilisation des classificateurs seuls n'a pas permis d'atteindre une grande précision. Cependant, la combinaison de SelectKBest avec PCA, RP ou LDA a entraîné des améliorations significatives de la précision. La meilleure précision de 100% a été obtenue en combinant SelectKBest avec des noyaux spécifiques de SVM et des techniques de réduction de dimension. En conclusion, cette recherche démontre l'efficacité des techniques de réduction de dimension, en particulier lorsqu'elles sont combinées à une technique de sélection de caractéristiques telle que SelectKBest dans notre cas, pour améliorer la précision de classification pour la détection des anomalies dans les mammogrammes.

Mots clés: images mammographiques, apprentissage, automatique, extraction de caractéristiques, sélection de caractéristiques, réduction de dimension, classification, LBP, SelectKBest, PCA, RP, SRP, GRP, LDA, KNN, SVM.

ملخص

سرطان الثدي هو سبب رئيسي للوفاة من السرطان عند النساء. الكشف المبكر والتشخيص ضروريان لتحسين معدلات البقاء على قيد الحياة. في هذه الدراسة، قمنا بتطوير نظام تشخيص بمساعدة الكمبيوتر (CAD) يعتمد على نهج التعلم الآلي للكشف عن سرطان الثدي. تم تقديم النظام المقترح للتحقيق في آثار تقنيات تقليل الأبعاد لتصنيف تصوير الثدي بالأشعة السينية. وهو يتألف من المعالجة المسبقة واستخراج الميزات واختيار الميزات وتقليل الأبعاد وخطوات التصنيف. يتم استخراج منطقة الاهتمام (ROI)، ويتم الحصول على الميزات التركيبية باستخدام الأنماط الثنائية المحلية (LBP). يتم استخدام SelectKBest كتقنية لاختيار الميزات لتحديد الميزات ذات الصلة بينما يتم تطبيق تقنيات تقليل الأبعاد، بما في ذلك تحليل المكونات الرئيسية (PCA)، والإسقاط العشوائي (RP)، والتحليل التمييزي الخطي (LDA)، لتقليل أبعاد الصور. يتم استخدام K-Nearest Neighbors (KNN) و Support Vector Machines (SVM)مع نواة Sigmoid وPolynomial وRadial Basis Function. الهدف هو مقارنة أداء تقنيات تقليل الأبعاد واختيار الميزات باستخدام هذين المصنفين. أظهرت النتائج الأولية أن استخدام المصنفات وحدها لم يحقق الدقة العالية. ومع ذلك، أدى الجمع بين SelectKBest وPCA أو RP أو LDA إلى تحسينات كبيرة في الدقة. تم تحقيق أفضل دقة بنسبة 100٪ عند الجمع بين SelectKBest ونواة محددة من SVM وتقنيات تقليل الأبعاد. في الختام، يوضح هذا البحث فعالية تقنيات تقليل الأبعاد، خاصةً عندما تقترن بتقنية اختيار الميزة التي هي SelectKBest في حالتنا، في تحسين دقة التصنيف للكشف عن التشوهات في تصوير الثدي بالأشعة السينية.

الكلمات المفتاحية: الصور الشعاعية للثدي، التعلم الآلي، استخراج الميزات، اختيار الميزات، تقليل البعد، التصنيف، LBP، PCA، RP، SRP، GRP، LDA، KNN، SVM.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

AGI. : *Artificial General Intelligence*

AI. : *Artificial intelligence*

ANN. : *Artificial Neural Network*

CAD. : *Computer-Aided Diagnosis*

CCA. : *Canonical Correlation Analysis*

CFS. : *Correlation-based Feature Selection*

CNN. : *Convolutional Neural Network*

DCIS. : *Ductal Carcinoma In Situ*

FN. : *False Negative*

FP. : *False Positive*

GA. : *Genetic Algorithms*

GRP. : *Gaussian Random Projection*

IBC. : *Inflammatory Breast Cancer*

ICA. : *Independent Component Analysis*

IDC. : *Invasive Ductal Carcinoma*

ILC. : *Invasive Lobular Carcinoma*

KNN. : *K-Nearest Nieghbors*

KPCA. : *Kernel Principal Component Analysis*

Lasso. : *Least Absolute Shrinkage and Selection Operator*

LBP. : *Linear Binary Patterns*

LCIS. : *Lobular Carcinoma In Situ*

LDA. : *Linear Discriminant Analysis*

MIAS. : *Mammographic Image Analysis Society*

ML. : *Machine learning*

OCR. : *Optical Character Recognition*

PCA. : *Principal Component Analysis*

PSO. : *Particle Swarm Optimization*

RBF. : *Radial Basis Function*

RFE. : *Recursive Feature Elimination*

RGB. : *Red, Green, Blue*

ROI. *See, : Region of Interest*

RP. : *Random Projection*

Sb. : Between-class scatter matrix

SBS. : *Sequential Backward Selection*

SFS. : *Sequential Forward Selection*

SNN. : *simulated neural network*

SRP. : *Sparse Random Projection*

SVD. : *Singular Value Decomposition*

SVM. : *Support Vector Machines*

Sw. : *Within-class scatter matrix*

TN. : *True Negative*

TNBC. : *Triple Negative Breast Cancer*

TP. : *True Positive*

t-SNE. : *t-Distributed Stochastic Neighbor Embedding*

WHO. *World Health Organization*

General Introduction

Breast cancer is a major public health concern posing a significant challenge in the medical field and early detection is essential for improving survival rates. Accurate diagnosis is crucial for effective treatment and improved patient outcomes. However, the interpretation of mammographic images, a key diagnostic tool for breast cancer, can be complex and subjective, leading to potential misdiagnosis or delayed treatment. To address this problem, our research aims to develop a computer-aided diagnosis (CAD) system based on machine learning techniques. This system offers a solution to improve the accuracy and efficiency of breast cancer detection by leveraging advanced image analysis algorithms and dimension reduction techniques. One of the main challenges in analyzing mammographic images is the high dimensionality of the feature space, which can hinder classification accuracy and computational efficiency. To overcome this challenge, we employ dimension reduction techniques such as Principal Component Analysis (PCA), Random Projection (RP), and Linear Discriminant Analysis (LDA). These techniques effectively reduce the dimensionality of the images while preserving important discriminatory information, enhancing the performance of subsequent classification algorithms. In addition to dimension reduction, we incorporate feature selection using SelectKBest, which helps in identifying the most informative features for breast cancer classification, reducing noise and improving the accuracy of the proposed CAD system. KNN and SVM are used in the classification task. By comparing the performance of different dimension reduction techniques and feature selection methods, we aim to identify the most effective combination for accurate breast cancer detection. By providing a robust and efficient CAD system for breast cancer detection, our research offers a valuable solution to overcome the challenges associated with the interpretation of mammographic images. The integration of advanced machine learning techniques, dimension reduction, and feature selection enables accurate and timely identification of abnormalities, contributing to early detection and improved patient outcomes in the fight against breast cancer.

Our work is structured into five chapters that focus on addressing the challenges of breast cancer detection and classification.

Chapter 1 (Breast cancer) provides an overview of breast cancer, its significance, and the challenges associated with accurate diagnosis.

Chapter 2 (Image processing and feature extraction) delves into image preprocessing techniques and feature extraction methods, with a specific emphasis on Local Binary Patterns (LBP).

Chapter 3 (Dimension reduction and feature selection) which represents our focus that shifts towards dimension reduction and feature selection. We investigate the effectiveness of dimension reduction techniques such as Principal Component Analysis (PCA), Random Projection (RP), and Linear Discriminant Analysis (LDA) in reducing the dimensionality of mammographic images while preserving crucial information. Additionally, we explore the utility of feature selection techniques, with a specific emphasis on SelectKBest, in identifying the most informative features for accurate classification.

Chapter 4 (Classification) focuses on classification algorithms, namely Support Vector Machines (SVM) and K-Nearest Neighbors (KNN), and evaluates their performance in classifying mammograms as normal or abnormal.

Chapter 5 (Results and discussion) presents the results obtained from our experiments and engage in a comprehensive discussion. We analyze the performance of the developed CAD system, considering the impact of dimension reduction techniques, feature selection methods, and classification algorithms. This chapter also highlights the strengths, limitations, and potential future directions of our research.

I. Chapter 1 Breast Cancer

1.1 Introduction

Breast cancer is a significant global health concern that affects millions of individuals, primarily women, and has far-reaching consequences for physical and emotional well-being. Understanding the current breast cancer statistics is vital in raising awareness, guiding prevention efforts, and enhancing early detection and treatment strategies. It is crucial to recognize that breast cancer not only affects women but also a smaller percentage of men. While the disease is more frequently diagnosed in women, men can also develop breast cancer. Although the incidence in men is significantly lower, accounting for less than 1% of all breast cancer cases, it is essential to raise awareness among both genders to ensure early detection and appropriate support.

Recent statistics underscore the alarming burden of breast cancer worldwide. According to the World Health Organization (WHO), breast cancer accounts for a substantial proportion of new cancer cases and cancer-related deaths among women. In 2020 alone, approximately 2.3 million new cases of breast cancer were diagnosed, representing about 11.7% of all new cancer cases. Moreover, breast cancer was responsible for approximately 685,000 deaths, ranking it as the fifth leading cause of cancer-related mortality on a global scale. These statistics emphasize the urgent need for increased awareness, early detection, and improved treatment options for breast cancer. Various factors contribute to the incidence and mortality rates of this disease, including age, genetics, lifestyle choices, and environmental influences. Identifying and addressing these risk factors play a pivotal role in reducing the burden of breast cancer on individuals and healthcare systems.

In this chapter, we explore breast cancer in depth, examining persons affected, types, symptoms, risk factors, stages, diagnostic techniques and treatment options. By staying informed and proactive, we can work towards reducing the burden of breast cancer, improving outcomes, and supporting individuals affected by this disease.

1.2 Definition

1.2.1 Cancer

Cancer is a group of diseases characterized by the uncontrolled growth and spread of abnormal cells in the body. It is caused by changes (mutations) in the DNA within cells, which disrupt the normal regulation of cell division and behavior. When cells don’t die at the normal rate, there’s more cell growth than cell death. This excess growth can form a tumor [1]. It can cause various symptoms depending on its type and location. Common signs and symptoms include the presence of a lump or thickening, changes in the skin, unexplained weight loss, fatigue, pain, changes in bowel or bladder habits, and abnormal bleeding. There are many different types of cancer, including **breast cancer**, lung cancer, prostate cancer, colorectal cancer, and leukemia, among others. Each type has its own characteristics, treatment options, and prognosis.

1.2.2 Breast

Breasts are made up of breast tissue (also called glandular tissue) and fat, along with nerves, veins, arteries, and connective tissue that helps hold everything in place. Breast tissue is a complex network of lobules (small round sacs that produce milk) and milk ducts (canals that carry milk from the lobules to the nipple openings during breastfeeding) in a pattern that looks like bunches of grapes. These bunches are called lobes [2].

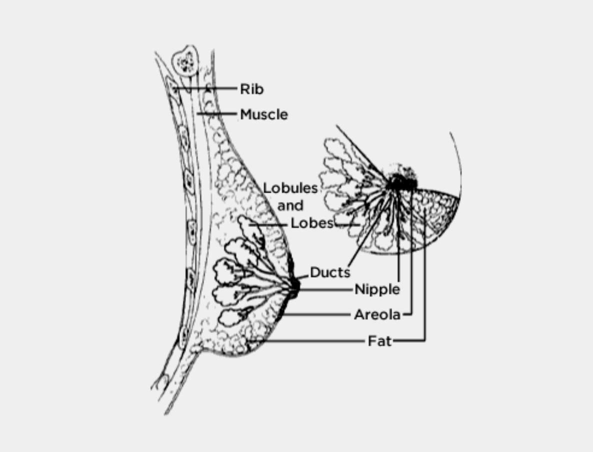


Figure I.1. Breast Structure [2]

1.2.2.1 The female breast

During childhood and adolescence, girls have a small patch of immature breast tissue. Puberty triggers hormonal changes that lead to breast growth, with milk ducts stretching and becoming more branched. Eventually, the breast tissue develops into mature lobules and ducts. In adulthood, women typically have 15-20 lobes in each breast, consisting of 20-40 lobules per lobe. The lobules are connected to small milk ducts, which gradually merge to form larger ducts. Each breast contains about 10 duct systems, each with its own opening at the nipple. The breast tissue remains inactive until pregnancy. During pregnancy, the lobules expand and begin producing milk, which is then released into the ducts for breastfeeding. Muscle tissue in the nipples allows them to become erect in response to stimulation or breastfeeding. Glands on the areola release fluid to lubricate the nipple during breastfeeding. After menopause, the number of lobules decreases, and the remaining lobules shrink in size. Breast tissue decreases, leading to a decrease in breast density. Before menopause, the breasts have a higher breast tissue-to-fat ratio, while after menopause, the breasts typically have more fat than breast tissue. This change in breast density can make it easier to interpret mammograms after menopause [2].

1.2.2.2 The male breast

In the early stages of life, both boys and girls possess similar breast tissue. However, unlike women, men do not experience the intricate growth and development of the breasts. During puberty, the presence of high levels of testosterone and low levels of estrogen in males halts breast development. While men do have some milk ducts, these ducts remain underdeveloped, and lobules are typically absent [2].

1.2.3 Breast cancer

Breast cancer develops within the breast tissue when certain cells undergo mutations and proliferate uncontrollably, resulting in the formation of a tissue mass known as a tumor. Similar to other types of cancer, breast cancer has the potential to invade and spread into the surrounding tissue of the breast [3]. Additionally, it can spread outside the breast through blood vessels and lymph vessels. When it spreads to other parts of the body, it is said to have metastasized [4].

1.2.3.1 Who is affected by breast cancer?

Breast cancer primarily affects women, although it can also occur in men, but much less frequently. Women of all ages are at risk of developing breast cancer, but the risk increases with age. The majority of breast cancer cases occur in postmenopausal women. Male breast cancer is very rare. Less than one percent of all breast cancer cases develop in men, and only one in a thousand men will ever be diagnosed with breast cancer. Breast cancer in men is usually detected as a hard lump underneath the nipple and areola. Men carry a higher mortality than women do, primarily because awareness among men is less and they are less likely to assume a lump is breast cancer [5]. For the age, breast cancer is most often diagnosed in adults over the age of 50, but it can occur at any age.

1.2.3.2 Early signs (Symptoms)

The symptoms of breast cancer can vary from person to person. Potential indications of breast cancer include:

- Changes in the size, shape, or contour of the breast.
- The presence of a mass or lump, which may be as small as a pea.
- The occurrence of a lump or thickening in or near the breast or underarm that persists throughout the menstrual cycle.
- Alterations in the appearance or texture of the skin on the breast or nipple, such as dimpling, puckering, scaliness, or inflammation.
- Redness of the skin on the breast or nipple.
- The presence of an area that significantly differs from other regions on either breast.
- The presence of a hardened area under the skin, resembling a marble-like texture.
- Discharge from the nipple, which can be blood-stained or clear fluid.

It is worth noting that some individuals may not experience any noticeable signs of breast cancer. This underscores the importance of regular mammograms and screenings as they can help detect breast cancer in its early stages, even when no apparent symptoms are present [3].

1.2.3.3 Risk factors

Research has indicated that the risk of developing breast cancer is influenced by a combination of factors. The primary factors that contribute to this risk are being female and advancing in age.

It's important to note that some women may develop breast cancer even in the absence of known risk factors. Having a risk factor does not guarantee that the disease will occur, and the impact of each risk factor can vary. While most women possess some risk factors, the majority do not develop breast cancer. If you have risk factors associated with breast cancer, it is advisable to consult with your doctor regarding strategies to mitigate your risk and the importance of regular breast cancer screening [6].

1.2.3.4 You can't change

- **Sex:** Women are much more likely to develop breast cancer than men [3].
- **Getting older:** In addition to gender, advancing age is a significant risk factor for breast cancer, as the incidence of the disease is closely correlated with increasing age. In 2016, a substantial proportion of breast cancer-related deaths in the United States, approximately 99.3% and 71.2%, were reported among women aged 40 and above, and 60 and above, respectively. Therefore, it is crucial for women aged 40 or older to undergo mammography screening in a timely manner [7].
- **Family history and genetic mutations:** Approximately 25% of breast cancer cases are associated with a family history of the disease. Women who have a mother or sister diagnosed with breast cancer are more susceptible to developing the disease [7]. Women who have inherited changes (mutations) to certain genes, such as BRCA1 and BRCA2, are at higher risk of breast and ovarian cancer [6].
- **Reproductive factors:** Reproductive factors, including early menarche, late menopause, late age at first pregnancy and low parity, can impact the risk of developing breast cancer. Every one-year delay in menopause raises the risk of breast cancer by 3%, while each one-year delay in the start of menstruation or each additional childbirth reduces the risk by 5% or 10% respectively [7].
- **Personal history of breast cancer or certain non-cancerous breast diseases:** Women with a history of breast cancer (who have had breast cancer) have an increased likelihood of experiencing a second occurrence of the disease. Certain non-cancerous

breast conditions, such as atypical hyperplasia or lobular carcinoma in situ, are also linked to a higher risk of developing breast cancer [6].

- **Having dense breasts:** Breasts with dense tissue contain a higher proportion of connective tissue compared to fatty tissue, which can occasionally pose challenges in detecting tumors on a mammogram. Women with dense breasts have an increased risk of developing breast cancer [6].
- **Radiation exposure:** If you’ve had prior radiation therapy — especially to your head, neck or chest — you’re more likely to develop breast cancer [3].

1.2.3.5 You can change (Causes)

- **Not being physically active:** Women who are not physically active have a higher risk of getting breast cancer [6].
- **Obesity:** Having obesity can increase your risk of breast cancer and breast cancer recurrence [3].
- **Smoking and drinking alcohol:** Tobacco use has been linked to many different types of cancer, including breast cancer [3]. The consumption of alcohol can raise the levels of estrogen-related hormones in the bloodstream and activate the estrogen receptor pathways. A comprehensive analysis combining 53 epidemiological studies revealed that consuming 35-44 grams of alcohol per day can elevate the risk of breast cancer by 32%. Additionally, each additional 10 grams of alcohol per day was associated with a 7.1% increase in the relative risk of developing breast cancer [7].
- **Taking hormones:** The use of certain hormone replacement therapies during menopause, specifically those containing both estrogen and progesterone, may increase the risk of breast cancer when used for more than five years. Similarly, specific types of oral contraceptives (birth control pills) have also been associated with an increased risk of breast cancer [6].
- **Reproductive history:** Having the first pregnancy after age 30, not breastfeeding, and never having a full-term pregnancy can raise breast cancer risk [6].

1.2.3.6 Diagnosis

Additional tests are frequently employed by doctors to detect or diagnose breast cancer. In some cases, women may be referred to specialized professionals, such as breast specialists or surgeons. It is important to note that such referrals do not necessarily indicate the presence of cancer or the need for surgery. Rather, these medical experts possess expertise in diagnosing various breast conditions and providing appropriate guidance and care [8]. Breast cancer can be diagnosed through multiple tests, including a mammogram, ultrasound, MRI, biopsy and Positron emission tomography (PET) scanning [9].

- **Mammogram:** Mammograms, which are specialized X-ray images, are effective in detecting changes or abnormal growths in the breast. They are commonly used as a preventive measure for breast cancer [3]. If you experience any issues with your breast, such as the presence of lumps or if an area appears abnormal on a screening mammogram, doctors may recommend a diagnostic mammogram. This type of mammogram provides a more detailed X-ray of the breast to further investigate the concern [8].

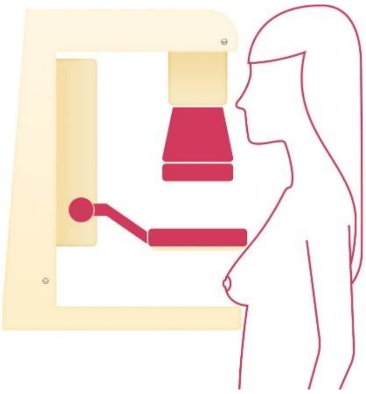


Figure I.2. Mammogram diagnosis [10]

- **Ultrasound:** If a suspicious area is identified in your breast during a breast self-exam or on a screening mammogram, your doctor may recommend a breast ultrasound. This imaging procedure utilizes sound waves that penetrate the breast tissue without causing any harm or radiation exposure. The echoes produced by the breast tissue interacting with the sound waves are captured by a computer to create a detailed picture of the internal breast tissue. This allows for differentiation between a liquid-filled mass and a solid mass [11].

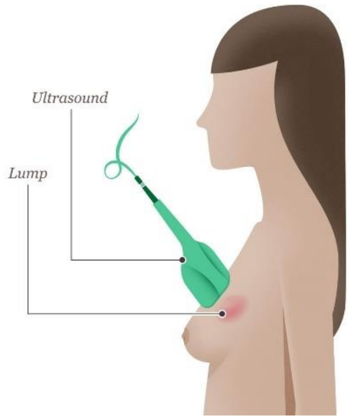


Figure I.3. Breast ultrasound [11]

- **Magnetic resonance imaging (MRI):** An MRI (magnetic resonance imaging) scan is a type of body scan that employs a magnet connected to a computer system. This imaging technique generates highly detailed pictures of the internal structures within the breast [8]. These images help the medical team distinguish between normal and diseased tissue.

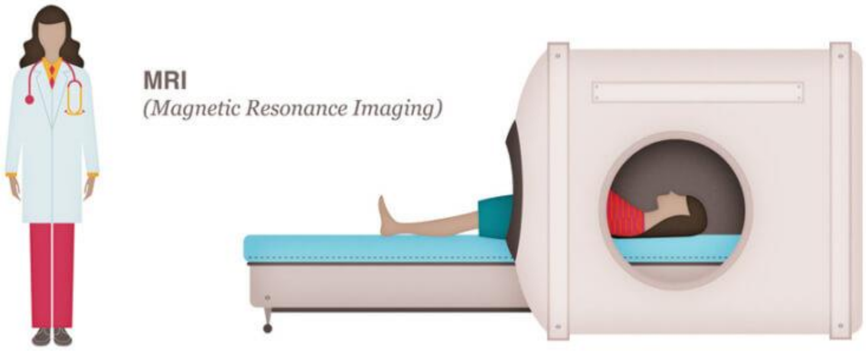


Figure I.4. Magnetic resonance imaging [12]

- **Biopsy:** A biopsy is a procedure that involves the extraction of a sample of cells or tissue from the area of concern in your breast. This sample is then carefully examined under a microscope to determine whether cancerous cells are present [3]. There are different kinds of biopsies (for example, fine-needle aspiration, core-needle biopsy, or surgical biopsy).

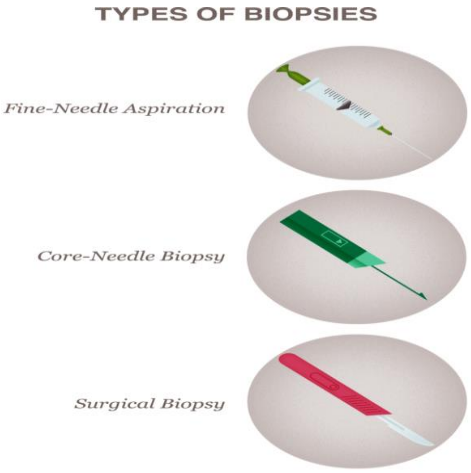


Figure I.5. Biopsies types [14]

- **Positron emission tomography (PET):** During a PET scan, specific dyes are utilized to enhance and identify areas of concern. In this procedure, a healthcare professional injects a specialized dye into your veins, and the scanner captures images to visualize the distribution of the dye in your body.

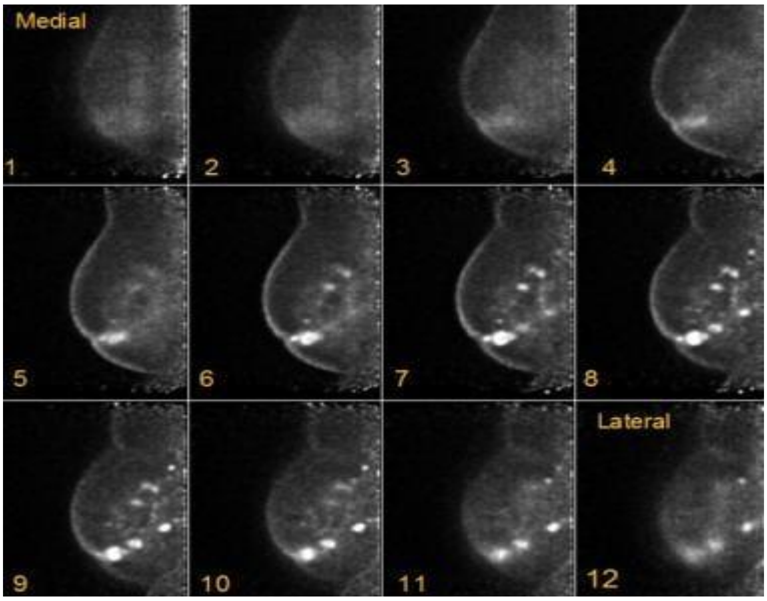


Figure I.6. multifocal infiltrating ductal carcinoma as seen on F-18 fluorodeoxyglucose positron emission tomography. This organ specific positron imaging technique results in a much higher resolution (down to 1 to 2 mm) compared to whole body PET/CT imaging. [15]

1.2.3.7 Stages

Staging is a crucial process in determining the extent of cancer spread within the breast and to other areas of the body. By gathering relevant information, the staging process determines the specific stage of the disease, which is essential for treatment planning. Some of the diagnostic tests used to detect breast cancer also provide valuable insights for staging the disease [16]. Staging provides valuable information about the size, location of the tumor, and the presence of metastasis. This process categorizes breast cancer into different stages, providing a comprehensive understanding of the disease [3]. The basic breast cancer stages are:

Table I.1. Breast Cancer Stages [17]

Stage	Description
Noninvasive	
0	No evidence of cancer cells or invasion of the basement membrane of the duct or neighboring normal tissue; includes ductal carcinoma in situ
Invasive	
IA	• Tumor ≤ 2 cm AND • No spread outside the breast; no lymph nodes involved
IB	• No tumor in the breast, but microscopic metastases (> 0.2 mm but ≤ 2 mm) present in axillary lymph nodes OR • Tumor present in the breast, ≤ 2 cm, with involvement of lymph nodes
IIA	• No tumor in the breast, but macroscopic cancer (> 2 mm) in 1-3 axillary lymph nodes OR • Tumor ≤ 2 cm, with spread to axillary lymph nodes OR • Tumor > 2 cm but ≤ 5cm, with no spread to axillary lymph nodes
IIB	• Tumor > 2 cm but ≤ 5cm, with spread to 1-3 axillary lymph nodes OR • Tumor > 5cm, with no spread to axillary lymph nodes
IIIA	• No tumor in the breast or presence of a breast tumor of any size associated with metastases in 4-9 axillary lymph nodes or in internal mammary nodes OR • Tumor > 5cm, with spread to axillary and/or internal mammary nodes
IIIB	• Tumor of any size, with spread to chest wall and/or skin of the breast; may also have spread to axillary or internal mammary nodes
IIIC	• Tumor of any size, with spread to ≥ 10 axillary lymph nodes OR • Spread to lymph nodes above or below the collarbone (supraclavicular nodes) OR • Spread to both axillary lymph nodes and internal mammary nodes
Metastatic	
IV	Spread of cancer to other parts of the body such as liver, lung or bone

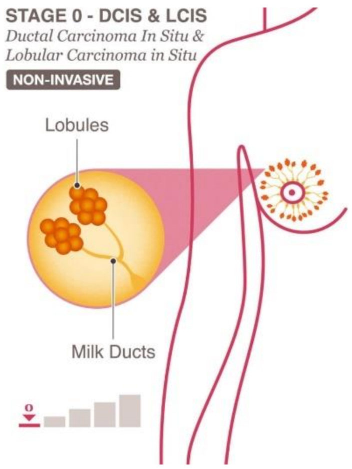


Figure I.7. Stage 0 [18]

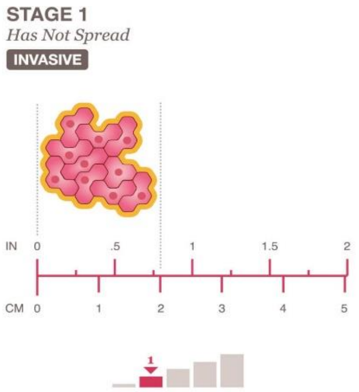


Figure I.8. Stage 1 [18]

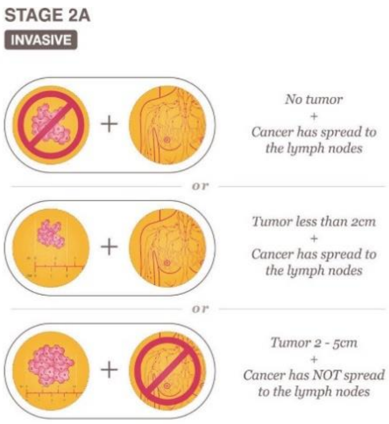

STAGE 2A
INVASIVE
No tumor
+
Cancer has spread to the lymph nodes
or
Tumor less than 2cm
+
Cancer has spread to the lymph nodes
or
Tumor 2 - 5cm
+
Cancer has NOT spread to the lymph nodes

Figure I.9. Stage 2A [19]

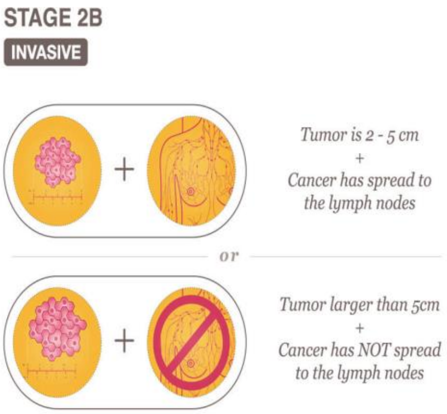

STAGE 2B
INVASIVE
Tumor is 2 - 5 cm
+
Cancer has spread to the lymph nodes
or
Tumor larger than 5cm
+
Cancer has NOT spread to the lymph nodes

Figure I.10. Stage 2B [19]

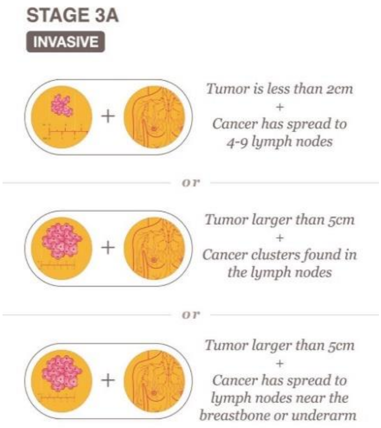

STAGE 3A
INVASIVE
Tumor is less than 2cm
+
Cancer has spread to 4-9 lymph nodes
or
Tumor larger than 5cm
+
Cancer clusters found in the lymph nodes
or
Tumor larger than 5cm
+
Cancer has spread to lymph nodes near the breastbone or underarm

Figure I.11. Stage 3A [20]

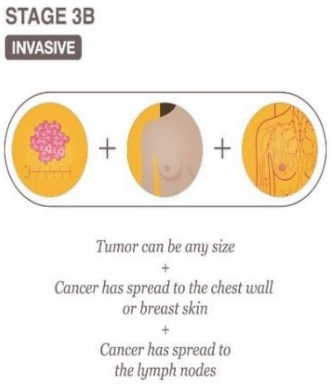

STAGE 3B
INVASIVE
Tumor can be any size
+
Cancer has spread to the chest wall or breast skin
+
Cancer has spread to the lymph nodes

Figure I.12. Stage 3B [20]

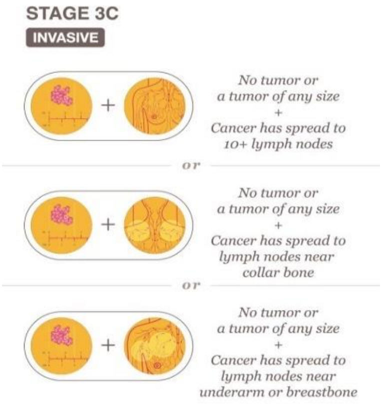

STAGE 3C
INVASIVE
No tumor or a tumor of any size
+
Cancer has spread to 10+ lymph nodes
or
No tumor or a tumor of any size
+
Cancer has spread to lymph nodes near collar bone
or
No tumor or a tumor of any size
+
Cancer has spread to lymph nodes near underarm or breastbone

Figure I.13. Stage 3C [20]

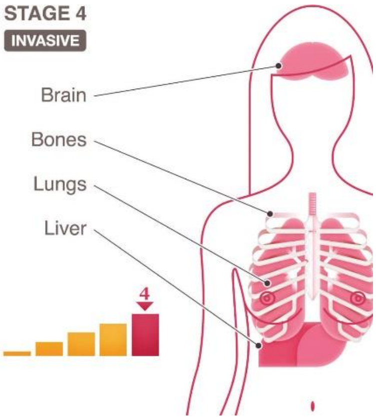


Figure I.14. Stage 4 [21]

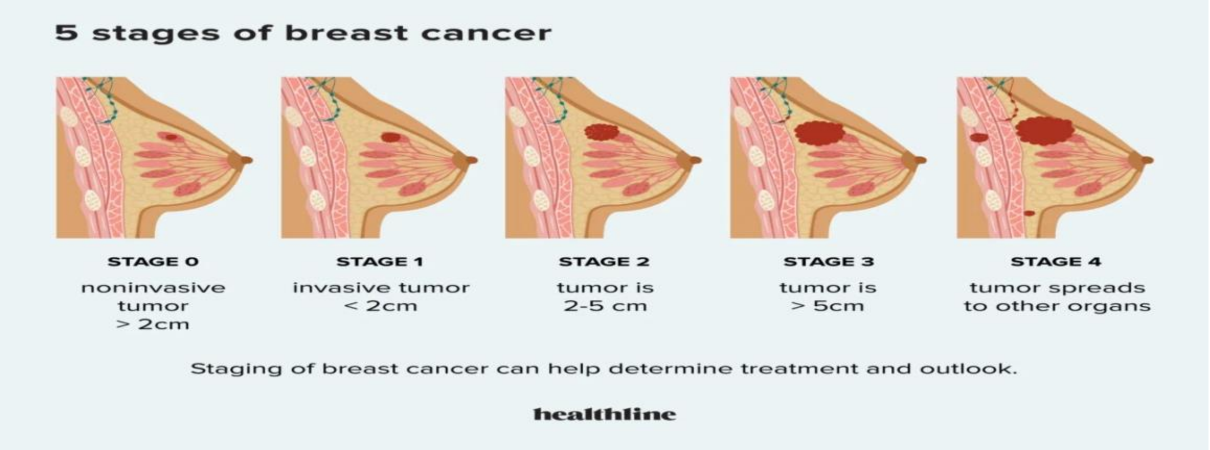

5 stages of breast cancer
STAGE 0
noninvasive tumor > 2cm
STAGE 1
invasive tumor < 2cm
STAGE 2
tumor is 2-5 cm
STAGE 3
tumor is > 5cm
STAGE 4
tumor spreads to other organs
Staging of breast cancer can help determine treatment and outlook.
healthline

Figure I.15. All stages [22]

1.2.3.8 Types of breast cancer

There are several different types of breast cancer, including:

- **Invasive Ductal Carcinoma (IDC):** Breast cancer originates from cells within the ducts and subsequently expands beyond the ducts, infiltrating various regions of the breast tissue. The invasive cancer cells possess the capability to metastasize, extending their reach to other distant parts of the body [4].

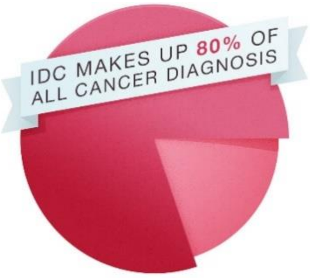


Figure I.16. Invasive Ductal Carcinoma [23]

- **Ductal Carcinoma In Situ (DCIS):** Ductal carcinoma in situ (DCIS) is a type of cancer that remains confined to the lining of the milk ducts in the breast. It is considered a non-invasive condition since the abnormal cells have not yet extended beyond the ducts into the surrounding breast tissue. DCIS is typically detected at an early stage and is highly treatable. However, if left untreated or undetected, there is a potential for it to progress and spread into the neighboring breast tissue [24].

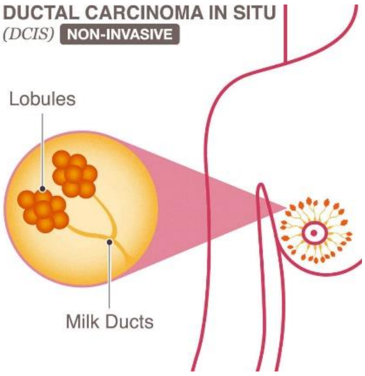


Figure I.17. DCIS [24]

- **Invasive Lobular Carcinoma (ILC):** This type of cancer develops within the lobules of the breast, which are responsible for producing breast milk, and has extended into the adjacent breast tissue. It represents approximately 10% to 15% of all breast cancer cases [3].

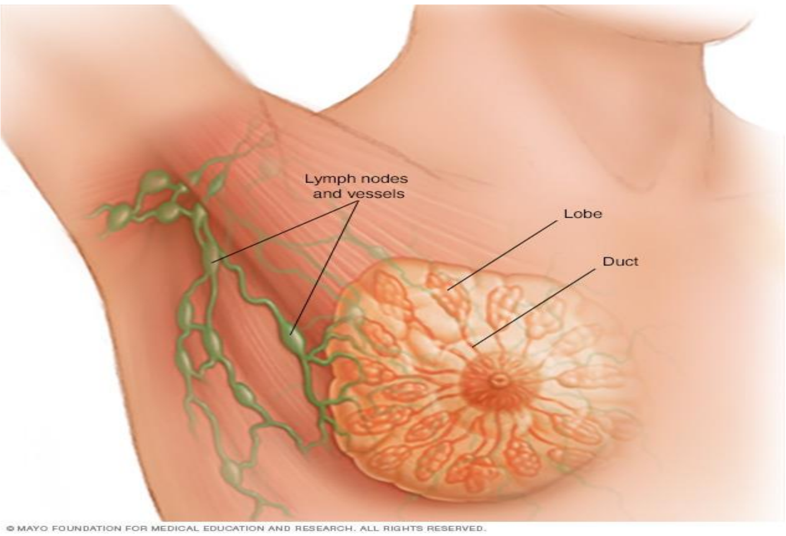
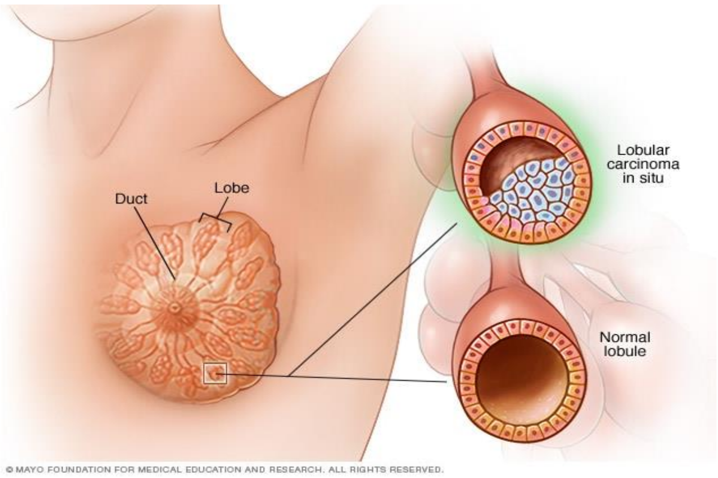

Lymph nodes and vessels
Lobe
Duct
© MAYO FOUNDATION FOR MEDICAL EDUCATION AND RESEARCH. ALL RIGHTS RESERVED.

Figure I.18. Invasive lobular carcinoma [25]

- **Lobular Carcinoma In Situ (LCIS):** Lobular Carcinoma In Situ (LCIS) refers to a condition characterized by the presence of abnormal cells within the lobules of the breast. These atypical cells have not yet disseminated beyond the confines of the lobules into the surrounding breast tissue [26].



© MAYO FOUNDATION FOR MEDICAL EDUCATION AND RESEARCH. ALL RIGHTS RESERVED.

Figure I.19. Lobular carcinoma in situ [27]

- **Triple Negative Breast Cancer (TNBC):** Triple negative breast cancer, constituting approximately 15% of all cases, presents a significant challenge in terms of treatment. It earns its name "triple negative" due to the absence of three markers commonly found in other types of breast cancer. This unique characteristic makes prognosis and treatment of this form of breast cancer particularly complex and challenging [3].

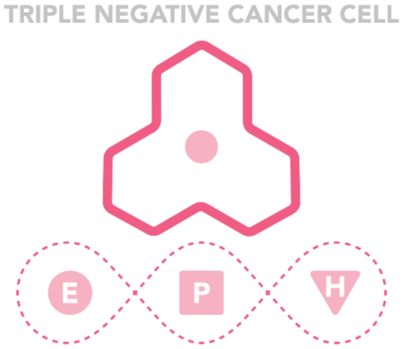


Figure I.20. TNBC [28]

- **Inflammatory Breast Cancer (IBC):** Inflammatory breast cancer is a highly aggressive and rapidly progressing form of breast cancer characterized by the infiltration of cancer cells into the skin and lymph vessels of the breast. This type of cancer often presents symptoms that resemble an infection, including redness, swelling, pitting, and dimpling of the breast skin. The underlying cause of these symptoms is the obstruction of the skin's lymph vessels by cancerous cells [3].

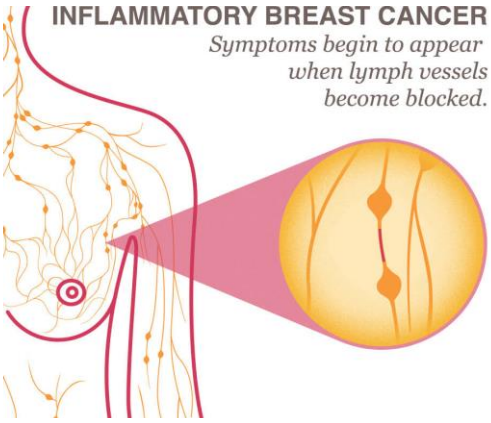

INFLAMMATORY BREAST CANCER
Symptoms begin to appear when lymph vessels become blocked.

Figure I.21. IBC [29]

- **Paget’s Disease Of The Breast:** This cancer affects the skin of your nipple and areola (the skin around your nipple) [3].

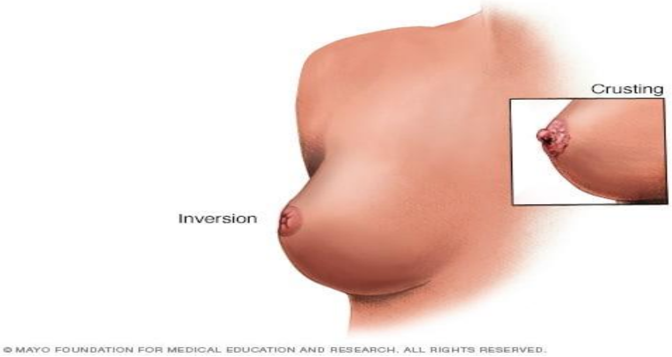

Crusting
Inversion
© MAYO FOUNDATION FOR MEDICAL EDUCATION AND RESEARCH. ALL RIGHTS RESERVED.

Figure I.22. Paget's disease of the breast [25]

- **Metastatic Breast Cancer:** The cancer has spread to other parts of the body. This usually includes the lungs, liver, bones or brain. The spread of cancer involves a series of steps. Initially, cancer cells invade and take over nearby healthy cells, causing them to replicate abnormally. Subsequently, these cancer cells have the ability to penetrate into the circulatory or lymph system by traversing the walls of adjacent lymph vessels or blood vessels. Through this mechanism, cancer cells can travel to other parts of the body. Once in circulation, they may lodge in capillaries at distant locations, where they divide and migrate into the surrounding tissue. This process leads to the formation of new small tumors, referred to as micro metastases. Thus, the spread of cancer involves a complex sequence of events that allows cancer cells to invade, travel, and establish secondary tumors in different areas of the body [30].

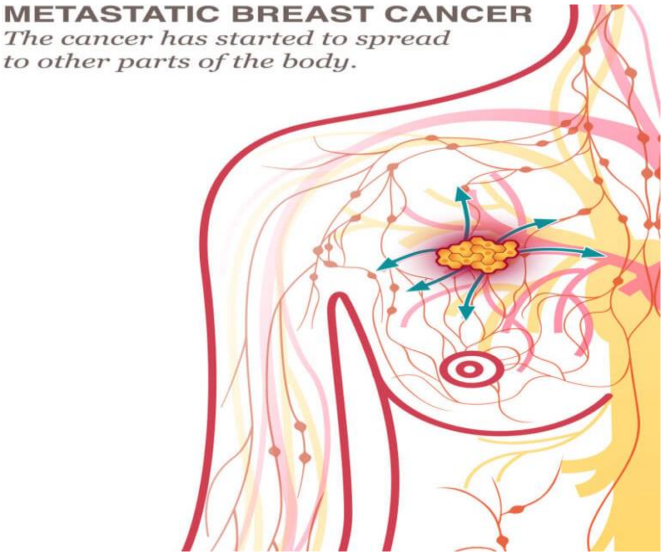

METASTATIC BREAST CANCER
The cancer has started to spread to other parts of the body.

Figure I.23. Metastatic breast cancer [30]

There are other types that are less commonly seen like Medullary Carcinoma and Tubular Carcinoma.

1.2.3.9 Breast Tumors

A tumor refers to an abnormal mass of tissue found in the body. Within the context of breast cancer, tumors can be classified into two main types: benign tumors, which are non-cancerous, and malignant tumors, which are cancerous [31].

1.2.3.9.1 Benign Tumors

Once a tumor is identified as benign, medical practitioners typically opt for a non-invasive approach, opting not to surgically remove it. Although benign tumors generally do not pose a significant threat to surrounding tissue, there are instances where they can gradually enlarge, exerting pressure on adjacent structures and leading to discomfort or complications. In such cases, the tumor is surgically excised, alleviating pain and resolving associated issues [31].

1.2.3.9.2 Malignant Tumors

Malignant tumors, being cancerous in nature, exhibit aggressiveness by infiltrating and causing harm to nearby tissues. When there is a suspicion of malignancy, a biopsy is conducted by the doctor to assess the tumor's degree of severity and aggressiveness [31].

1.2.3.10 Treatment

Breast cancer treatment involves various approaches that depend on the type and stage of the cancer. Typically, individuals with breast cancer undergo a combination of different treatments to effectively manage the disease [32]. Typically, breast cancer treatment plans encompass a range of options, and most commonly involve a combination of the following five approaches: surgery, radiation therapy, hormone therapy, chemotherapy, and targeted therapies. These treatments can be either local, focusing on the specific area surrounding the tumor, or systemic, designed to target the entire body using agents that combat cancer [33].

- **Choosing your doctor:** Once you receive a breast cancer diagnosis, you will collaborate with a multidisciplinary team of cancer specialists, which may consist of the following healthcare professionals:
    - A medical oncologist
    - A surgical oncologist
    - A radiation oncologist
    - A care manager, caseworker, nurse navigator, or patient navigator

Figure I.24. Cancer team specialists [34]

Additionally, your healthcare team may comprise an oncology nurse and a registered dietitian [34].

- **Surgery:** Surgery is the primary treatment method for breast cancer, and it typically involves the removal of the tumor and surrounding tissue. Surgical options for breast cancer may include a lumpectomy, partial mastectomy, radical mastectomy, or reconstruction procedures [35]. An operation where doctors cut out cancer tissue [32]. Types of surgery include the following:

- **Breast-conserving surgery:** Breast-conserving surgery is an operation to remove the cancer and some normal tissue around it, but not the breast itself. Part of the chest wall lining may also be removed if the cancer is near it. This type of surgery may also be called lumpectomy, partial mastectomy, segmental mastectomy, quadrantectomy, or breast-sparing surgery [36].

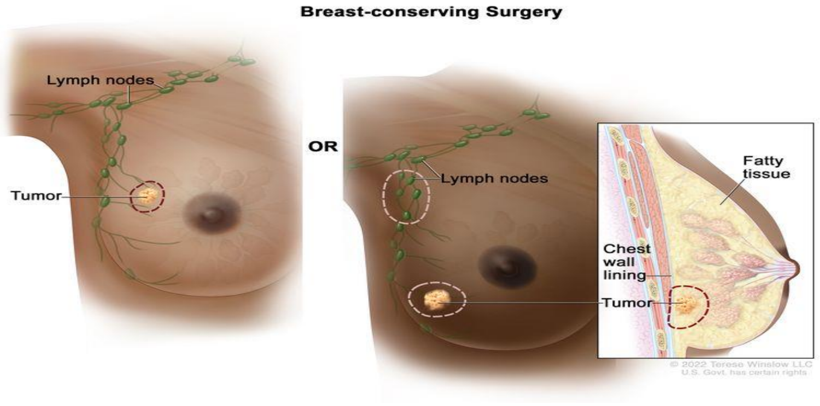

Breast-conserving Surgery
Lymph nodes
Tumor
OR
Lymph nodes
Fatty tissue
Chest wall lining
Tumor
© 2022 Terese Winslow LLC U.S. Govt. has certain rights

Figure I.25. Breast cancer surgery [36]

- **Total mastectomy:** Total mastectomy, also known as a simple mastectomy, is a surgical procedure that involves the complete removal of the breast affected by cancer. In addition, some of the lymph nodes in the underarm area may be removed and examined for the presence of cancer cells. This lymph node evaluation can be performed either during the same surgery as the breast removal or as a separate procedure, using a distinct incision [36].

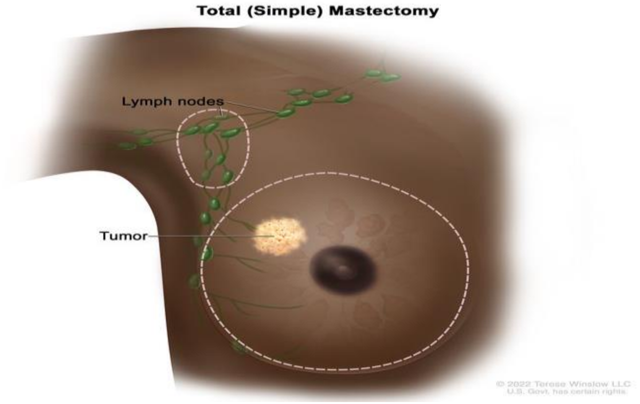

Total (Simple) Mastectomy
Lymph nodes
Tumor
© 2022 Terese Winslow LLC U.S. Govt. has certain rights

Figure I.26. Total Mastectomy [36]

- **Modified radical mastectomy:** Modified radical mastectomy is a surgical procedure that involves the complete removal of the breast affected by cancer. This may entail the removal of the nipple, areola (the pigmented area surrounding the nipple), and the overlying skin of the breast. Additionally, the majority of the lymph nodes located under the arm are also surgically removed [36].

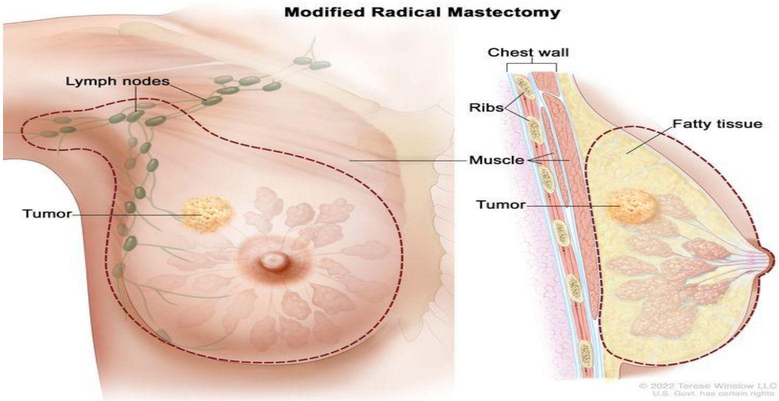

Modified Radical Mastectomy
Lymph nodes
Tumor
Chest wall
Ribs
Muscle
Tumor
Fatty tissue
© 2022 Terese Winslow LLC
U.S. Govt. has certain rights

Figure I.27. Modified radical mastectomy [36]

- **Chemotherapy:** In some cases, your healthcare provider may suggest chemotherapy as a preliminary treatment for breast cancer before undergoing a lumpectomy, aiming to reduce the size of the tumor. Alternatively, chemotherapy may be administered after surgery to eliminate any residual cancer cells and decrease the likelihood of recurrence. If the cancer has metastasized to other areas of the body, your healthcare provider may advise chemotherapy as the primary treatment option [3].

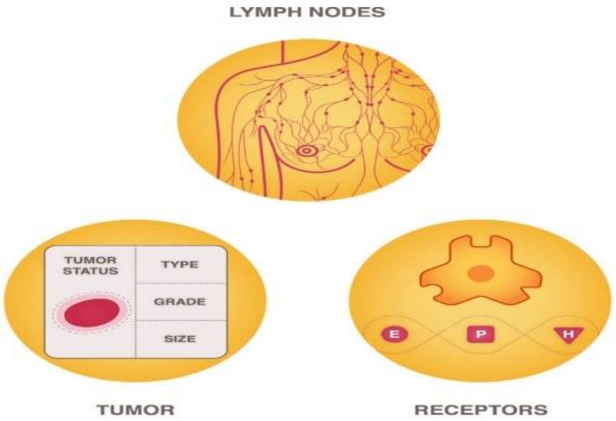


Figure I.28. Chemotherapy [37]

- **Radiation therapy:** Radiation therapy, also known as radiotherapy, employs powerful beams of high-energy rays to eliminate cancer cells. It specifically targets the cells within the treated area of the body. In the case of breast cancer, radiation therapy is utilized to eradicate any residual abnormal cells that may persist in the breast or armpit region following surgery. There are two types of radiation therapy:

  - **External radiation therapy:** uses a machine outside the body to send radiation toward the area of the body with cancer [36].
  - **Internal radiation therapy:** uses a radioactive substance sealed in needles, seeds, wires, or catheters that are placed directly into or near the cancer [36].

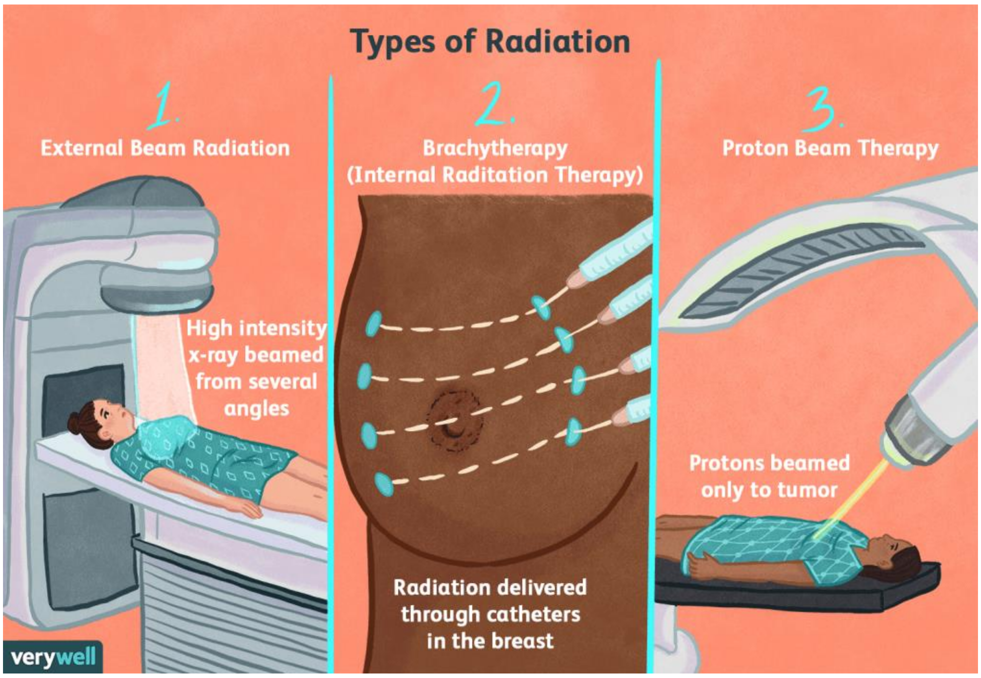

Types of Radiation
1.
External Beam Radiation
High intensity x-ray beamed from several angles
2.
Brachytherapy (Internal Raditation Therapy)
Radiation delivered through catheters in the breast
3.
Proton Beam Therapy
Protons beamed only to tumor
verywell

Figure I.29. Radiation therapy types [38]

- **Hormonal therapy:** Blocks cancer cells from getting the hormones they need to grow [32].

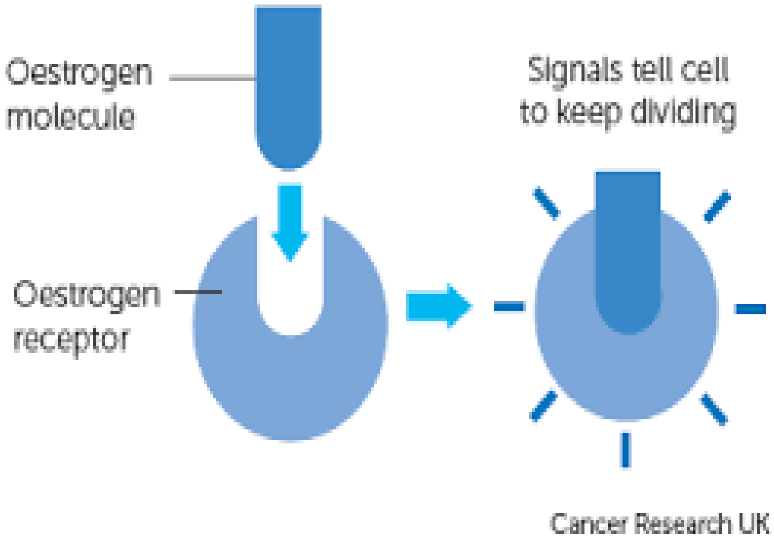


Figure I.30. Hormonal therapy [39]

- **Targeted therapy:** Alongside chemotherapy and hormone therapy, there exist contemporary treatments that exhibit enhanced effectiveness by selectively targeting specific breast cancer cells while sparing normal cells. Presently, these targeted approaches are frequently employed alongside conventional chemotherapy. Notably, targeted medications often yield milder side effects compared to traditional chemotherapy drugs [40].

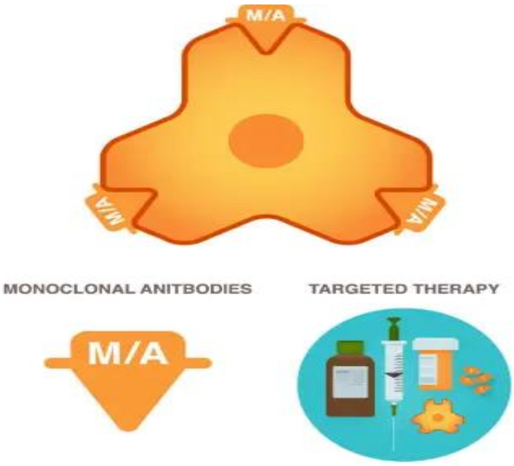


Figure I.31. Targeted therapy [40]

1.3 Conclusion

Breast cancer is a complex and multifaceted disease that affects a significant number of individuals, predominantly women. It is crucial to be aware of the risk factors and symptoms associated with breast cancer, as early detection and diagnosis can greatly improve the chances of successful treatment and favorable outcomes. Various diagnostic procedures, including mammograms, ultrasounds, and biopsies, play a crucial role in determining the stage and extent of the disease. Treatment options for breast cancer encompass a range of approaches, such as surgery, radiation therapy, chemotherapy, hormone therapy, and targeted therapies. A comprehensive and multidisciplinary approach involving a team of healthcare professionals ensures that each patient receives personalized and optimal care. Ongoing research and advancements in medical technology continue to contribute to the development of more effective and targeted treatments for breast cancer. Furthermore, raising awareness, promoting regular screenings, and supporting further research are vital in the ongoing battle against breast cancer and the pursuit of improved outcomes and quality of life for those affected by the disease.

Next, we’re going to talk about how images are preprocessed and how relevant patterns and features are extracted from these images. In particular, how to deal with mammograms.

II. Chapter 2 Image Processing and Feature Extraction

2.1 Introduction

Human perception heavily relies on images, as vision is our most advanced sense. However, unlike humans who perceive only the visual band of the electromagnetic spectrum, imaging machines can work across a wide range of wavelengths, including gamma and radio waves. They can process images from sources that are not traditionally associated with visual information, such as ultrasound, electron microscopy, and computer-generated images. This diversity of sources contributes to the expansive field of applications within digital image processing.

The boundaries between image processing, image analysis, and computer vision lack consensus among authors. Some define image processing as a discipline where both the input and output of a process are images. However, this definition can be limiting and artificial. For example, a simple operation like computing the average intensity of an image, resulting in a single number, would not be considered an image processing operation under this definition. On the other hand, computer vision aims to emulate human vision using computers, including learning, making inferences, and taking actions based on visual inputs. As a branch of artificial intelligence (AI), computer vision seeks to replicate human intelligence but has experienced slower progress than initially expected.

Image analysis, or image understanding, resides between image processing and computer vision. The boundaries within the continuum from image processing to computer vision are not precisely defined. Nonetheless, it can be helpful to conceptualize three types of computerized processes along this continuum: low-level, mid-level, and high-level processes. Low-level processes involve fundamental operations like image preprocessing to reduce noise, enhance contrast, and sharpen images. These processes take images as inputs and yield images as outputs. Mid-level processing includes tasks such as segmentation, which partitions an image into regions or objects, as well as describing and classifying those objects. While mid-level processes typically take images as inputs, they extract attributes from the images, such as edges, contours, and object identities. Higher-level processing encompasses making sense of recognized objects within an ensemble, as seen in image analysis, and ultimately performs cognitive functions associated with vision.

In this chapter, we will delve into the fascinating world of image processing and feature extraction. Our goal is to provide a comprehensive understanding of image fundamental concepts such as what is an image, image types etc... ,to explore how features are extracted and what are the techniques used for feature extraction and what are the applications involved in this field.

2.2 Image

An image can be defined as a two-dimensional function, denoted as f(x, y), where x and y represent spatial coordinates, and the amplitude of f at any given (x, y) pair is referred to as the image's intensity or gray level at that point. If x, y, and the intensity values of f are discrete and finite, the image is considered as a **digital image**. Digital image processing involves the manipulation of digital images using a digital computer. It is important to note that a digital image consists of a finite number of elements, each having a specific location and value. These elements are commonly referred to as picture elements, image elements, pels, or pixels, with the term "pixel" being the most widely used to describe the elements of a digital image [41].

2.2.1 Image types

The images types we will consider are: binary, gray-scale, color, and multispectral.

1. Binary images

Binary images are the most basic form of images that are characterized by having only two possible values, typically represented as black and white or 0 and 1. A binary image is often referred to as a 1-bit image because each pixel can be represented using a single binary digit. These types of images are commonly used in applications that require only general shape or outline information, such as optical character recognition (OCR).

Binary images are frequently derived from grayscale images through a process called thresholding. In this operation, each pixel in the grayscale image is compared to a threshold value. If a pixel's intensity is above the threshold, it is assigned the value of white ('1'), and if it is below the threshold, it is assigned the value of black ('0'). This conversion results in a binary image where pixels above the threshold appear white and pixels below, the threshold appear black [42].

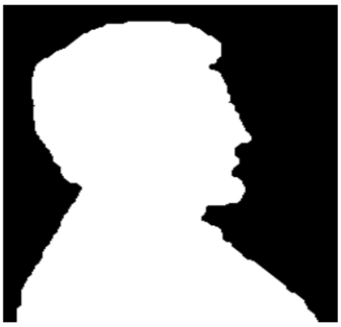


Figure II.1. Binary image Object outline [42]

2. Gray-Scale images

Grayscale images, also referred to as black and white images, are images where each pixel is represented by a single intensity value that corresponds to a specific shade of gray. Unlike color images that consist of multiple color channels, grayscale images contain only one channel, which represents the brightness or intensity of each pixel.

The intensity values in grayscale images typically span a range from 0 (representing black) to 255 (representing white) in an 8-bit representation. Darker shades of gray correspond to lower intensity values, while lighter shades correspond to higher intensity values. Intermediate intensity values represent various shades of gray between the extremes of black and white.

Grayscale images find wide applications in image processing and analysis tasks due to their ability to simplify the representation of visual information by eliminating color information. They prove particularly valuable when color is unnecessary for the specific task at hand or when the focus lies on analyzing intensity variations and textures within the image.

To obtain a grayscale image from a color image, the RGB (Red, Green, Blue) channels of the color image are converted into a single grayscale channel. This conversion merges the color information from the three channels into a single intensity value for each pixel, resulting in a grayscale rendition of the original image.

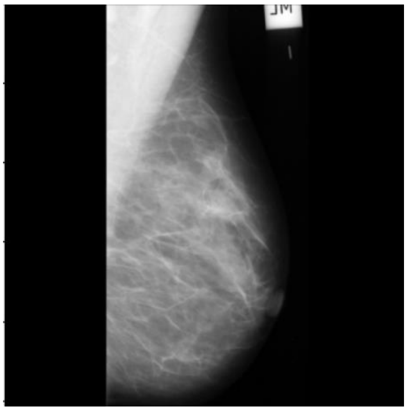


Figure II.2. mdb044 from MIAS dataset [43]

3. Color images

Color images are typically composed of three-band monochrome image data, where each band represents a different color. Each spectral band contains gray-level information that represents the actual data stored in the digital image [44].

Commonly, color images are represented in the RGB (Red, Green, Blue) format. Following the 8-bit monochrome standard, a corresponding RGB color image would consist of 24 bits per pixel, allocating 8 bits for each of the three-color bands: red, green, and blue [42].

Figure II.3. Color image

4. Multispectral images

These images encompass information that lies beyond the conventional human perceptual range. As a result, the represented data is not directly visible to the human visual system, which differentiates them from traditional images. Nevertheless, by mapping various spectral bands to the RGB components, the information is transformed into a visual form that can be interpreted by humans. Multispectral images encompass a wide range of data, including ultraviolet, infrared, X-ray, radar data, and acoustic signals [44].

2.2.2 Image characteristics

Image characteristics refer to the visual properties and attributes that can be analyzed and quantified in an image. These characteristics provide valuable information about the content, quality, and structure of an image. Some common image characteristics include :

1. **Pixel intensity**

Pixel intensity represents the brightness or color value of a pixel. In grayscale images, pixel intensity ranges from 0 (black) to 255 (white) in an 8-bit representation. In color images, each color channel (red, green, blue) has its intensity value, typically ranging from 0 to 255. Pixel intensity provides information about the brightness and color composition of an image.

2. **Noise**

Image noise refers to random fluctuations or variations in the brightness or color information of captured images. It is a form of degradation in the image signal that can be caused by various external factors. When it comes to multiplicative noise, brighter areas in the image tend to exhibit higher levels of noise. However, in most cases, image noise is considered to be additive, meaning it is independent of the pixel intensity and is present uniformly across the image [45].

Common types of noise include Gaussian noise (random variations following a Gaussian distribution), salt-and-pepper noise (sporadic occurrence of black and white pixels), and speckle noise (granular pattern caused by interference). Noise reduction techniques aim to minimize these distortions and improve image quality [45].

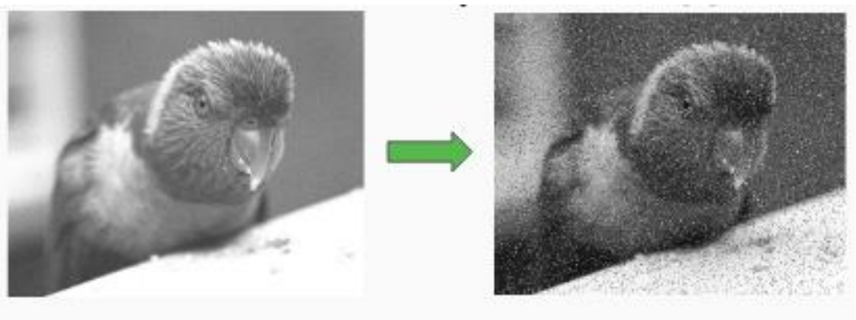


Figure II.4. Addition of random noise in an image [45]

3. Histogram

A histogram is a visual representation of the distribution of pixel intensities in an image. It displays the number of pixels at each different intensity value present in the image. In an 8-bit grayscale image, there are 256 possible intensity values, and the histogram will depict 256 values, indicating how the pixels are distributed among these grayscale levels. Histograms can also be generated for color images, either by obtaining individual histograms for the red, green, and blue channels, or by creating a 3-D histogram where the three axes represent the red, green, and blue channels, and the brightness at each point represents the pixel count [46].

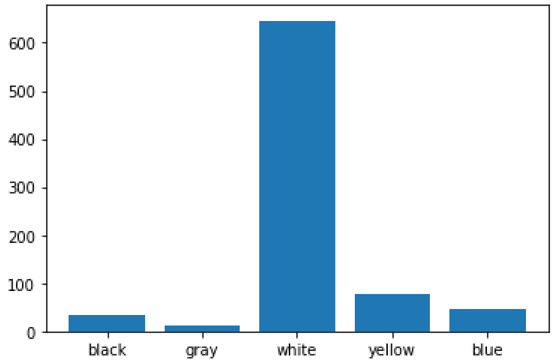


Figure II.5. Histogram visualized as bar plot [47]

4. Texture

Texture refers to the presence of patterns or repetitive structures in an image, which are characterized by variations in pixel intensities within specific regions. These patterns can exhibit qualities such as smoothness, roughness, coarseness, or specific arrangements like lines, dots, or grids. Analyzing texture enables us to understand the spatial organization of pixels and can be advantageous in tasks such as image classification, object recognition, and texture synthesis.

5. Homogeneity

Homogeneity refers to the consistency or uniformity of pixel intensities within specific regions in an image. In a homogeneous image, the pixel values show minimal variation or change either within a localized region or across the entire image.

When an image exhibits high homogeneity, it appears smooth and lacks abrupt changes or variations in pixel intensities. This quality is often desirable in applications like medical imaging, where the presence of uniform pixel intensities can indicate a healthy or regular structure.

Conversely, an image with low homogeneity displays significant variations or abrupt changes in pixel intensities, resulting in a more textured or heterogeneous appearance. This characteristic becomes valuable in tasks such as image segmentation, as it allows for the differentiation of different regions or objects based on their varying pixel intensities.

Figure II.6. Color homogeneity [48]

Still there are more characteristics which are represented in:

1) Sharpness which refers to the level of detail and clarity in an image.
2) Contrast that measures the difference in pixel intensities between the darkest and brightest regions of an image.
3) Resolution which refers to the level of detail captured in an image.
4) Geometric features that denote to spatial properties of objects or regions within an image.
5) Color features that describe characteristics related to the color information in an image.
6) Frequency content which refers to the distribution and magnitude of different frequencies present in an image.

2.2.3 Image enhancement

Image enhancement involves enhancing the interpretability or perception of information in images for human viewers and optimizing the input for automated image processing techniques. The main goal of image enhancement is to adjust the characteristics of an image to make it more appropriate for a particular task and a specific observer. Image enhancement aims to improve the quality and suitability of the image for its intended purpose [49].

Image enhancement techniques can be broadly categorized into two types: spatial domain enhancement and frequency domain enhancement.

1) **Spatial domain enhancement:** This type of enhancement operates directly on the pixels of the image in the spatial domain. Common techniques include :
    a. Histogram equalization: Adjusts the distribution of pixel intensities to enhance contrast.
    b. Brightness and contrast adjustment: Modifies the overall brightness and contrast of the image.
    c. Sharpening: Enhances edge details and improves overall image sharpness.
    d. Noise reduction: Removes or reduces unwanted noise in the image.
2) **Frequency domain enhancement:** This type of enhancement involves transforming the image into the frequency domain using techniques like the Fourier Transform. Common techniques include :
    a. Filtering: Enhances or suppresses certain frequency components to emphasize or remove specific image features.
    b. Homomorphic filtering: Adjusts the illumination and reflectance components of the image separately to enhance details.
    c. Wavelet transform: Decomposes the image into different frequency sub bands and allows enhancement at different scales.

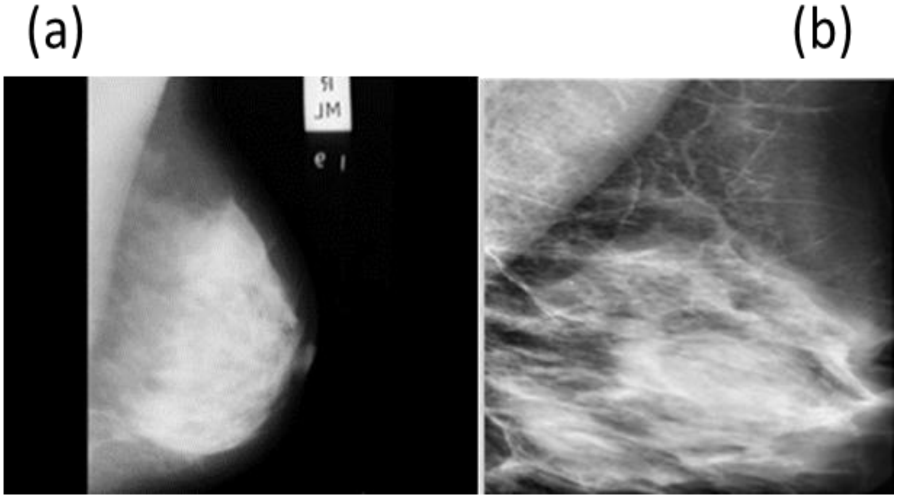


Figure II.7. (a) Original image (b) Enhanced image [50]

2.3 Preprocessing

Mammogram images encompass various types of information that are undesirable for the algorithm and have the potential to mislead the classification process. Additionally, mammograms often exhibit poor clarity and low contrast. Consequently, a preprocessing phase becomes imperative to enhance the image quality and establish a more reliable foundation for the subsequent feature extraction phase [50]. First, we used MIAS images as they are. Then, we used (LBP) as a feature extraction technique. Finally, the LBP is applied to the ROI (Region Of Interest) which was found by removing the background, labels and the pectoral muscle.

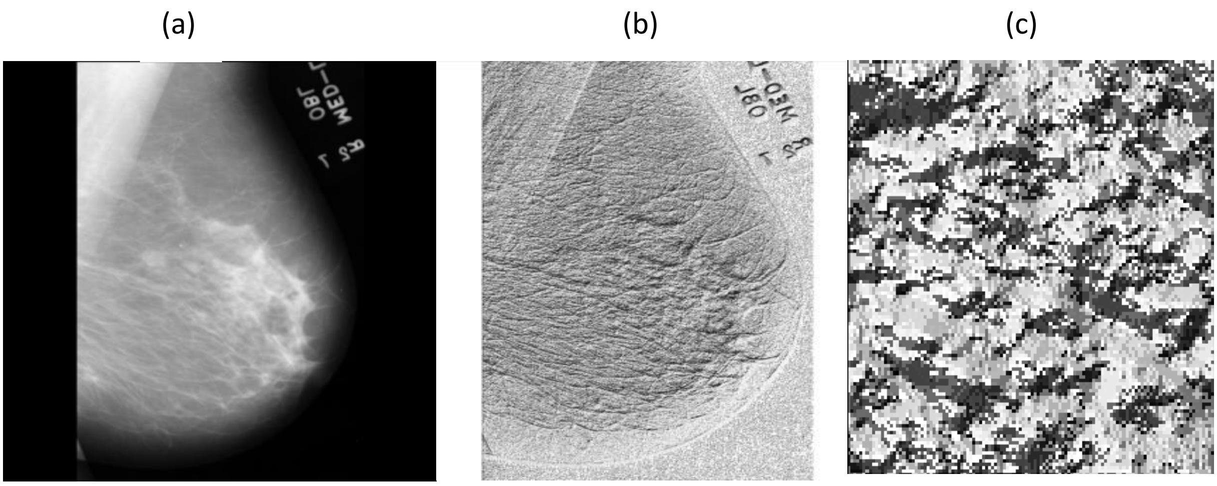


Figure II.8. mdb082 from MIAS dataset (a) full image (b) image with LBP (c) ROI image [43]

Feature extraction involves generating a set of features, often represented as vectors, that describe the content of an image and serve as a representation of the image itself. These extracted features play a crucial role in classification tasks. In the case of mammogram images, which contain diverse and heterogeneous information representing various tissues, the feature extraction phase becomes essential. Its purpose is to reduce the dimensionality of the mammogram images, transforming them into a reduced set of representative features [50].

There are various types of features that can be extracted from mammogram images, depending on the specific analysis task and requirements. Some commonly used types of features in mammogram image processing include:

- **Global features:** Global features consider the entire image or larger regions to extract information about the overall characteristics. These features provide a holistic representation of the image and are often used for image-level analysis [51]. Examples of global features include:

  - Color histograms: Histograms that represent the distribution of pixel colors or color channels in an image. They capture the color composition and can be useful for tasks such as color-based image retrieval or image classification.
  - Statistical moments: Measures such as mean, variance, skewness, or kurtosis that capture statistical properties of pixel intensities in an image. These moments provide information about the image's overall brightness, contrast, or distribution.
  - Fourier descriptors: Features obtained by applying the Fourier transform to an image or its regions. They capture frequency information and are useful for shape analysis and recognition tasks.

- **Deep learning features:** Deep learning techniques, particularly convolutional neural networks (CNNs), have a strong ability to extract complex features that express the image in much more detail, learn the task specific features and are much more efficient [52]. Features can be extracted from intermediate layers of a CNN, such as fully connected layers or convolutional feature maps:

1. Activation-based features: Activations from intermediate layers can serve as features, representing learned patterns and structures in the image.
2. Convolutional feature maps: Feature maps produced by the convolutional layers of a CNN capture hierarchical representations of visual information.

- **Local features:** A local descriptor refers to a representation of a specific patch or region within an image. Using multiple local descriptors for image matching provides increased resilience compared to relying solely on a single descriptor. By considering multiple descriptors, the matching process becomes more robust and capable of capturing diverse characteristics and variations present within the image [51]. Examples of local features include:
- Corners: Points in an image where two or more edges meet. Corner detection algorithms, such as the Harris corner detector, can identify these distinctive points.
- Edges: Areas of rapid intensity transition in an image. Edge detection algorithms, such as the Canny edge detector, can identify and highlight these boundaries.
- **Texture patterns:** Descriptive characteristics of small patches in an image, such as the presence of fine-grained textures, roughness, or smoothness. **Local Binary Patterns (LBP)** is a popular texture descriptor used for feature extraction.

2.4.1 Local Binary Patterns

The local binary pattern (LBP) is a texture image descriptor that characterizes the local spatial structure and contrast of an image or a specific region within it. LBP has gained significant popularity as a texture descriptor due to its simplicity in implementation, ability to extract informative features with high classification accuracy, and widespread adoption by researchers in the field. A notable advantage of the LBP method is its robustness in handling uniform changes in grayscale values, making it a highly suitable approach for various image analysis tasks. Additionally, LBP offers computational efficiency, further contributing to its effectiveness and practicality in real-world applications [53]. The LBP operator assigns labels to pixels in an image by comparing the center value with the thresholded values of its neighborhood, typically a 3x3 region. The result of this comparison is treated as a binary number. Once all pixels have been labeled with their respective LBP codes, a histogram of these labels is computed and used as a texture descriptor. To compute the LBP code of a pixel, it is compared to its neighboring pixels [54].

the LBP operator as shown in figure 9 is done as the following:

- It uses eight neighbor pixels and considers the center value as a threshold.
- Then, it compares the center value with its neighbors. If the center value is greater than the neighbor's value, the operator generates 1. Otherwise, it generates 0.
- The output gives 8 binary numbers that represent the feature vector (Which is usually converted to decimal)

$$LBP_{\mathrm{P,R}} = \sum_{P=0}^{P-1} S(g_{\mathrm{P}} - g_0)2^P \; S(x) = \begin{cases} 0, & x < 0 \\ 1, & x \geq 0 \end{cases} \qquad (1)$$

Where, g_0 is the gray level value of the central pixel, g_{P} is the value of its neighbors. P is the number of involved neighbors [54].

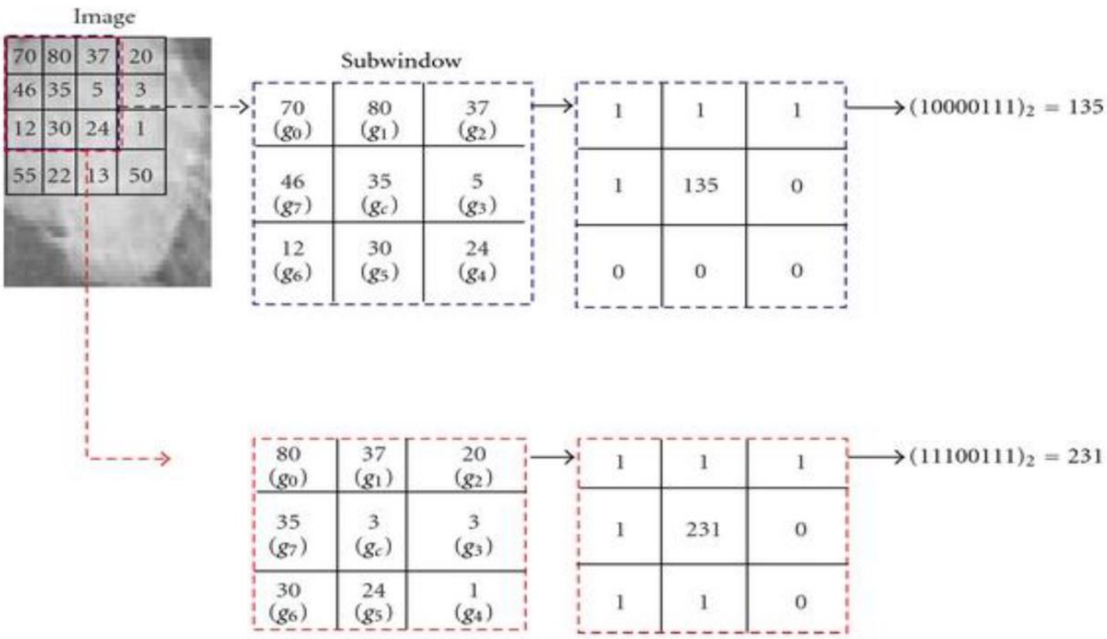


Image
70 80 37 20
46 35 5 3
12 30 24 1
55 22 13 50
Subwindow
70 (g_0) 80 (g_1) 37 (g_2)
46 (g_7) 35 (g_c) 5 (g_3)
12 (g_6) 30 (g_5) 24 (g_4)
1 1 1
1 135 0
0 0 0
$(10000111)_2 = 135$
80 (g_0) 37 (g_1) 20 (g_2)
35 (g_7) 3 (g_c) 3 (g_3)
30 (g_6) 24 (g_5) 1 (g_4)
1 1 1
1 231 0
1 1 0
$(11100111)_2 = 231$

Figure II.9. LBP operation [55]

We used the LBP technique for feature extraction for many reasons such as:

- Robustness to monotonic grey scale changes and robust against illumination changes
- Covers a small area of the neighborhood on specific radius.
- Works efficiently for singling out visual significant data
- Describes each pixel by the relative grey its neighboring pixels
- Few parameters required
- Fast computational ability

2.5 Applications of image processing and feature extraction

Feature extraction has a wide range of applications across various fields. Some common applications of feature extraction include:

- Object Recognition and Detection by extracting discriminative features, such as edges, corners, or local descriptors.
- Biometric Identification systems such as face recognition, fingerprint recognition, or iris recognition.
- Data Visualization by reducing high-dimensional data into a lower-dimensional space while preserving its essential characteristics. Extracted features are used to create meaningful visual representations that aid in data exploration and analysis.

- Medical Imaging in tasks like tumor detection, lesion segmentation, and disease classification. Texture features, shape features, or intensity-based features are extracted to characterize abnormal regions in medical images and aid in diagnosis and treatment planning.
- Image Classification by training machine learning models to distinguish between different classes of images. Features representing distinct patterns, textures, or shapes can be extracted and used as input for classification algorithms.

2.6 Conclusion

Image processing and feature extraction are essential components that unlock the vast potential of digital images, allowing us to extract valuable insights, make informed decisions, and deepen our understanding of the visual world. Through continuous expansion of knowledge and expertise in this field, we actively contribute to the progress of computer vision, artificial intelligence, and various domains that heavily rely on visual information. This collective effort pushes the boundaries of what can be achieved in image analysis and interpretation, leading to innovative applications and advancements in the field.

In the upcoming chapter, we will delve into the importance of dimension reduction and feature selection which complement the concepts of image processing and feature extraction by providing techniques and methodologies for managing high-dimensional data and selecting the most relevant features for analysis.

III. Chapter 3 Dimension reduction and feature selection

3.1 Introduction

As scientific disciplines and technological advancements progress, the volume and complexity of scientific data continue to expand. One challenge posed by this complexity is the presence of numerous covariates, making it difficult to discern the relationship between a response variable and the covariate set. To address the issue of numerous covariates, the statistical literature offers two main approaches. The first approach is variable selection which is also known as feature selection, where researchers believe that only a few covariates are truly associated with the response variable, while the rest are redundant and lack explanatory power.

The second approach is known as dimension reduction. Unlike feature selection, dimension reduction assumes that the response variable is related to a small number of linear combinations of the covariates. It is possible that all the covariates have some explanatory effect, but this effect is encapsulated within a few linear combinations. The objective of dimension reduction is to identify these critical linear combinations.

In this chapter, we explore dimension reduction and feature selection, which play a vital role in data analysis and machine learning. As datasets grow in size and complexity, it becomes crucial to extract valuable insights from them. Dimension reduction and feature selection techniques help us overcome this challenge by identifying the most important features and reducing the data's dimensionality. We explore unsupervised dimension reduction methods such as Principal Component Analysis (PCA), Random Projection (RP), and supervised techniques like Linear Discriminant Analysis (LDA). These techniques allow us to discover hidden patterns and compress high-dimensional data into a lower-dimensional representation, facilitating visualization and clustering. Additionally, we navigate the world of feature selection algorithms, including filter, wrapper, and embedded methods. These algorithms select or rank features based on their predictive power or relevance to the target variable. We explore one filter method which is SelectKBest. By mastering feature selection and dimension reduction techniques, navigating with high-dimensional data will become easy.

3.2 Feature selection

Feature selection is a crucial step in predictive modeling, aiming to reduce the number of input variables. By doing so, it achieves computational efficiency and, in some cases, improves model performance. Statistical-based methods assess the relationship between each input variable and the target variable using appropriate statistics. The selected variables are those that exhibit the strongest relationship with the target variable. However, choosing the right statistical measure depends on the data types of both input and output variables, posing a challenge for practitioners during filter-based feature selection [56]. Feature selection is a critical aspect of machine learning and data analysis, involving the identification and selection of **relevant features** from a given dataset. It aims to reduce the dimensionality of the data by selecting a subset of features that are most informative and impactful for the learning task.

3.2.1 Relevant feature

In the machine learning literature, the definition of "relevance" for features can vary depending on the specific question or goal at hand. Different definitions may be more appropriate in different scenarios. In this context, we explore various definitions of relevance and their significance. By doing so, we aim to shed light on the complexities, motivations, and diverse approaches found in the literature [57]. We can define feature relevance in many ways such as:

- **Relevant to the target:** A feature xi is relevant to a target concept c if there exists a pair of examples A and B in the instance space such that A and B differ only in their assignment to Xi and c(A) # c(B) [57]. In other words, it’s the degree to which that feature provides meaningful information or predictive power about the target variable we are trying to predict or classify. A relevant feature is one that is informative and helps improve the accuracy or performance of the machine learning model in predicting the target variable.

- **Relevant to the sample/distribution:** A feature Xi is considered highly relevant to a sample S when there are two examples, A and B, within S that differ solely in terms of their values for Xi and possess different labels. Similarly, Xi is deemed strongly relevant to the target variable c and the underlying distribution D when there are examples A and B that have a non-zero probability within D, differ only in their values for Xi, and satisfy the condition c(A) ≠ c(B) [57]. A feature's relevance to the sample or distribution is determined by assessing its ability to differentiate or capture important characteristics of the data points within that sample or distribution. Relevant features are those that exhibit significant variations or patterns across different samples or instances. In other words, a relevant feature should possess discriminative properties that allow it to distinguish between different classes or groups within the dataset. It should contribute to the understanding and representation of the underlying data distribution.

- **Feature relevance as a complexity measure:** Feature relevance can be viewed as a complexity measure in machine learning. It quantifies the importance or significance of features in a dataset and serves as a metric to assess the complexity of the data representation. As the number of features increases, the complexity of the dataset also tends to rise. Irrelevant or redundant features introduce noise and unnecessary complexity, which can lead to overfitting and reduced model generalization. On the other hand, relevant features capture meaningful patterns and characteristics, contributing to a more concise and informative representation of the data [57].
- **Feature relevance as incremental usefulness:** In the context of a given dataset S and a learning algorithm L, a feature Xi is considered incrementally useful to L when its inclusion in the feature set A improves the accuracy of the hypothesis generated by L compared to using A alone. To elaborate, if we consider the feature set A and then add the feature Xi to form the set {xi} U A, the hypothesis produced by L using this augmented feature set should exhibit higher accuracy than the hypothesis generated using only the feature set A. This concept of incremental usefulness is particularly relevant for feature-selection algorithms that iteratively explore different feature subsets by incrementally adding or removing features from their current set. These algorithms aim to identify the most valuable features by systematically evaluating their impact on the model's accuracy. By assessing the incremental usefulness of features, we can refine and enhance the feature set used by the learning algorithm, ultimately leading to improved model performance and predictive accuracy [57].

3.2.2 Methods of feature selection

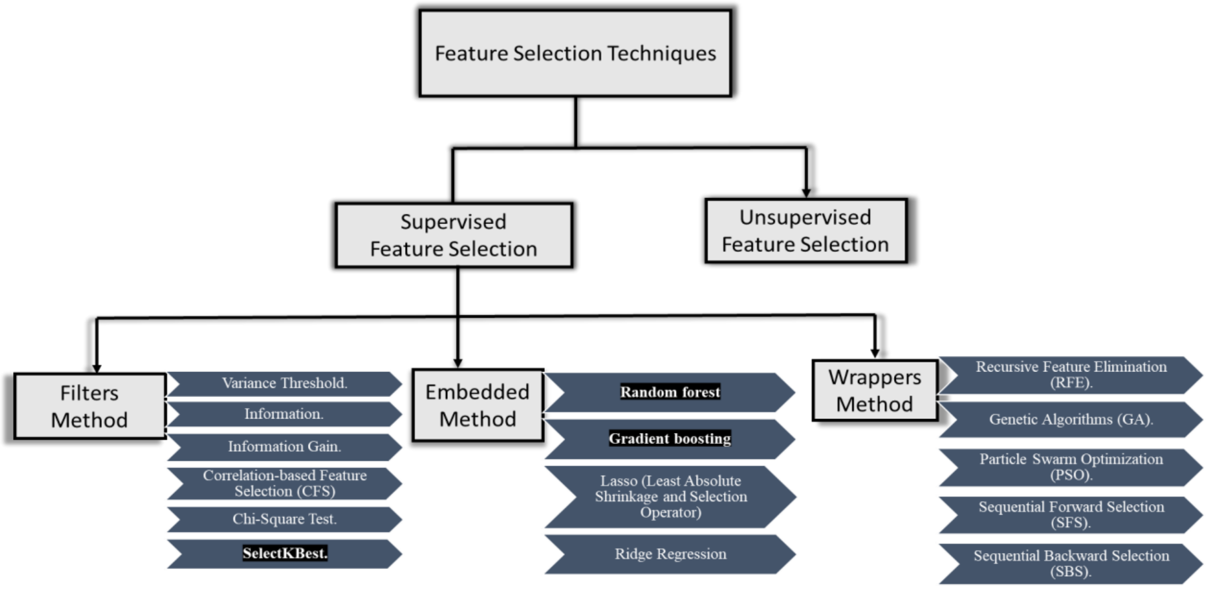

Feature Selection Techniques
Supervised Feature Selection
Unsupervised Feature Selection
Filters Method
Variance Threshold.
Information.
Information Gain.
Correlation-based Feature Selection (CFS)
Chi-Square Test.
SelectKBest.
Embedded Method
Random forest
Gradient boosting
Lasso (Least Absolute Shrinkage and Selection Operator)
Ridge Regression
Wrappers Method
Recursive Feature Elimination (RFE).
Genetic Algorithms (GA).
Particle Swarm Optimization (PSO).
Sequential Forward Selection (SFS).
Sequential Backward Selection (SBS).

Figure III.1. Methods of Feature Selection

In predictive modeling, dealing with a large number of variables presents various difficulties such as the complexity of model development, extended training time, and increased memory demands. Additionally, the inclusion of irrelevant input variables can negatively impact the performance of the models [56].

To overcome these challenges, feature selection methods offer a solution by identifying and selecting the most informative variables. These methods can be categorized into two broad groups: **supervised** and **unsupervised**.

- **Unsupervised feature selection methods**: Unsupervised feature selection is generally designed for clustering problems. These methods focus only on the characteristics and relationships among the input variables. They aim to identify the most meaningful and distinctive features in the dataset, without explicitly considering the target variable (e.g., removing redundant variables) (correlation).
- **Supervised feature selection methods:** Supervised feature selection is generally designed for classification or regression problems**.** These methods consider the relationship between the input variables and the target variable. By assessing the relevance of each variable to the target, these methods aim to choose the subset of features that contribute the most to predictive accuracy and model performance (e.g.,

remove irrelevant variables). In supervised methods we find 3 categories of methods which are:

- **Wrapper methods:**

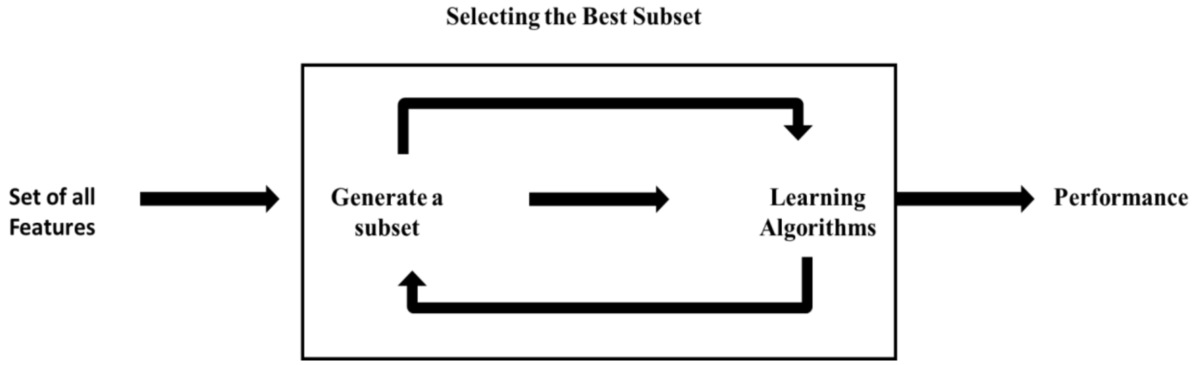


Figure III.2 Wrapper Methods process

Wrapper methods rely on the predictive performance of a pre-defined learning algorithm to assess the quality of selected features. Typically, a wrapper method consists of two steps: (1) searching for a subset of features and (2) evaluating the selected features. The tow processes (1) and (2) iterate until certain stopping criteria is met. In the feature set search step, the method generates various subsets of features, and the learning algorithm acts as a black box to evaluate the quality of each subset based on its performance. The process continues iteratively until the highest learning performance is achieved or the desired number of features is obtained. The subset of features that yields the best learning performance is then selected. In other words, wrapper methods search for well-performing subsets of features.

However, a major drawback of wrapper methods is that the search space grows exponentially with the number of features (2^d), making it impractical for datasets with a large number of features. To address this issue, different search strategies have been proposed, including sequential search, hill-climbing search, best-first search, branch-and-bound search, and genetic algorithms. These strategies aim to find a local optimum in terms of learning performance within the vast search space. Nonetheless, even with these strategies, the search space remains extremely large for high-dimensional datasets, limiting the practical applicability of wrapper methods [58]. While wrapper methods may have limitations in terms of computational complexity, there are still several popular wrapper methods that have been widely used in practice. Some of these methods include:

  - Recursive Feature Elimination (RFE).
  - Genetic Algorithms (GA).
  - Particle Swarm Optimization (PSO).
  - Sequential Forward Selection (SFS).
  - Sequential Backward Selection (SBS).

- **Filter methods:**

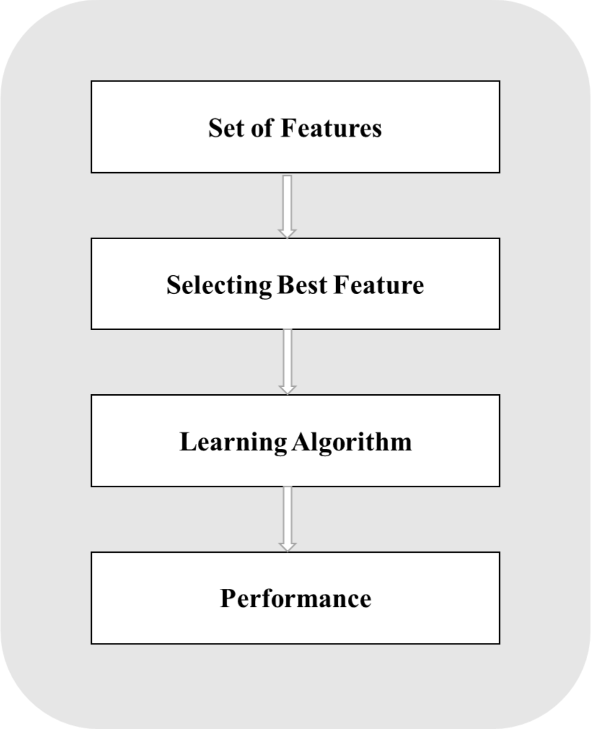


Figure III.3 Filter Methods Process

Filter methods are a type of feature selection technique that operate independently of any specific learning algorithm. They assess the importance of features based on the characteristics of the data. Compared to wrapper methods, filter methods are generally more computationally efficient. However, since they do not consider a particular learning algorithm during feature selection, the selected features may not be optimal for the target learning algorithms.

A typical filter method consists of two steps. In the first step, features are ranked based on a specific feature evaluation criterion. This evaluation process can be performed in either a univariate or multivariate manner. In the univariate scheme, each feature is ranked individually, irrespective of other features. In the multivariate scheme, multiple features are ranked simultaneously. In the second step of a filter method, features with low rankings are filtered out. Over the past decades, various evaluation criteria have been proposed for filter methods. These criteria aim to capture different aspects of feature importance. Examples include the feature's discriminative ability in separating samples, feature correlation, mutual information, preservation of data manifold structure, and ability to reconstruct the original data [58]. The filter method selects subsets of features based on their relationship with the target.

By applying different evaluation criteria, filter methods enable the identification of relevant features that exhibit desirable characteristics according to the specific evaluation criteria employed. These methods have proven useful in various domains for reducing feature dimensionality and enhancing the efficiency and interpretability of predictive models. Filter methods are categorized into two groups which are: statistical methods and feature importance methods. Some of the most commonly used filter methods are:

- Variance Threshold.
- Mutual Information.
- Information Gain.
- Correlation-based Feature Selection (CFS)
- Chi-Square Test.
- **SelectKBest**.

- **Embedded methods:**

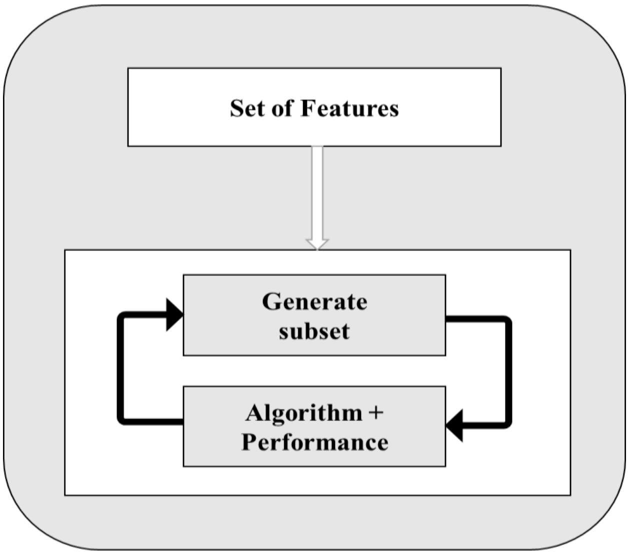


Figure III.4 Embedded Methods Process

Embedded methods are a tradeoff between filter and wrapper methods by integrating feature selection within the model learning process. This approach combines the advantages of both wrapper and filter methods. Firstly, embedded methods consider the interactions with the learning algorithm, leveraging the predictive power of the model itself. Secondly, they are more computationally efficient compared to wrapper methods as they do not require iterative evaluation of feature subsets. One of the most commonly used embedded methods is regularization. Regularization models aim to fit a learning model by minimizing the errors while simultaneously enforcing small or zero coefficients for certain features. These models effectively identify the most relevant features and return both the regularization model and the selected feature set as the final outcomes of the feature selection process [58]. Several embedded methods are used for feature selection such as:

- Random forest
- Gradient boosting
- Lasso (Least Absolute Shrinkage and Selection Operator)
- Ridge Regression

In summary, Wrapper methods evaluate the performance of a specific learning algorithm by iteratively selecting different subsets of features. They consider the interactions between features and the learning algorithm but can be computationally expensive. Filter methods, on the other hand, assess feature importance based on data characteristics using statistical measures. They are computationally efficient but do not consider the interactions with the learning algorithm. Embedded methods strike a balance between wrapper and filter methods by integrating feature selection into the model learning process. They leverage the interactions between features and the learning algorithm, resulting in efficient and effective feature selection. The choice of method depends on factors such as computational resources, the specific learning algorithm, and the desired level of interpretability.

3.2.2.1 *SelectKBest*

SelectKBest is widely recognized as a popular and frequently employed approach for feature selection in machine learning. It falls under the category of filter-based methods, which prioritize features based on statistical tests such as chi-squared, ANOVA F-test, or mutual information score. By assessing the relationship between each feature and the output variable, SelectKBest assigns scores and ranks them accordingly. Subsequently, it selects the top K features with the highest scores to form the final subset of features. SelectKBest offers convenience and simplicity in its implementation, enabling the rapid reduction of the feature set to a more manageable size. This attribute proves particularly advantageous when working with extensive datasets [59].

3.2.2.1.1 SelectKBest parameters

The SelectKBest feature selection method in scikit-learn has two main parameters that are:

- **Score function:** This parameter specifies the scoring function to evaluate the relationship between features and the target variable. Some common choices include chi-squared (chi2), **ANOVA F-test** (**f_classif** for classification tasks, f_regression for regression tasks), and mutual information (mutual_info_classif for classification tasks, mutual_info_regression for regression tasks). It possible to define your custom scoring function.
- **K:** This parameter determines the number of top features to select based on their scores. Set k to an integer value, or you can use other methods to automatically determine the value of k, such as selecting a percentile of features.

There are additional optional parameters that can be used with SelectKBest, such as score_func_kwds to pass additional arguments to the scoring function, and percentile to select features based on a percentile threshold rather than specifying a specific number of features.

3.2.2.1.2 SelectKBest's work

Here's a step-by-step overview of how SelectKBest works:

- The user has to choose the scoring function to use which defines the measure used to evaluate the relationship between each feature and the target variable, depending on the problem. For example, Anova F-test (f_classif) for classification or (f_regression) for regression task.
- SelectKBest applies the chosen scoring function to each feature individually. It computes a score that quantifies the feature's relevance or importance with respect to the target variable. The scoring function considers the statistical properties of the feature and its association with the target variable.
- After scoring all the features, SelectKBest ranks them based on their scores. Features with higher scores are considered more relevant or informative for predicting the target variable.
- SelectKBest selects the top K features specified by the user with the highest scores from the ranking. These selected features form the final feature subset.

- Finally, SelectKBest transforms the original dataset by retaining only the selected features and discarding the rest. This transformed dataset consists of the reduced feature subset, which can then be used for further analysis or model training.

For the Anova F-test statistic, the formula is:

$$F = \frac{explained\ variance}{unexplained\ variance}\ , \tag{2}$$

or

$$F = \frac{between-group\ variability}{within-group\ variability}\ . \tag{3}$$

The explained variance, or between-group variability is:

$$\sum_{i=1}^{k} n_i(\bar{y}_i - \bar{y})^2 / K - 1 \tag{4}$$

Where $\bar{y}_i$ denotes the sample mean in the i-th group, n_i is the number of observations in the i-th group, $\bar{y}$ denotes the overall mean of the data, and K denotes the number of groups.

The unexplained variance, or within-group variability is:

$$\sum_{i=1}^{k} \sum_{j=1}^{n_i} (y_{ij} - \bar{y}_i)^2 / N - K \tag{5}$$

Where yij is the j^{th} observation in the i^{th} out of K groups and N is the overall sample size. This F-statistic follows the F-distribution with degrees of freedom $d_1 = K - 1$ and $d_2 = N - K$ under the null hypothesis. The statistic will be large if the between-group variability is large relative to the within-group variability, which is unlikely to happen if the population means of the groups all have the same value.

3.2.3 Benefits and drawbacks of feature selection

Table III.1. Benefits and drawbacks of feature selection

Benefits	Drawbacks
Improved performance of machine learning models	Dependency on Feature Representation and Preprocessing
Reduced dimensionality	Computational complexity
Improved interoperability of the data	Sensitivity to Feature Selection Technique
Faster Training and Inference	Challenges in Handling Feature Interactions
Reduced Overfitting	Potential Information Loss

Overall, feature selection is a process in machine learning that aims to select a subset of relevant features from a larger set of input variables. It offers several benefits. But there are some drawbacks to consider. The choice of the best technique for the task to complete is necessary to effectively leverage feature selection and optimize model performance.

3.3 Dimension reduction

Dimensionality reduction is a valuable technique employed to decrease the number of features in a dataset while retaining the essential information. Its primary goal is to transform high-dimensional data into a lower-dimensional representation that captures the underlying structure of the original data.

In the field of machine learning, high-dimensional data refers to datasets with numerous features or variables. Dealing with high-dimensional data poses challenges due to the curse of dimensionality. As the number of features increases, the performance of models tends to degrade. This occurs because the complexity of the model grows with the number of features, making it more difficult to find optimal solutions. Additionally, high-dimensional data can lead to overfitting, where models become excessively tailored to the training data and struggle to generalize well to new data.

Dimensionality reduction techniques address these challenges by reducing the model's complexity and enhancing its ability to generalize. Two main approaches are commonly employed: feature selection, which selects a subset of relevant features, and feature extraction, which transforms the original features into a new set of lower-dimensional features capturing the essential information. These approaches provide efficient means to handle high-dimensional datasets and improve the performance of machine learning models [60].

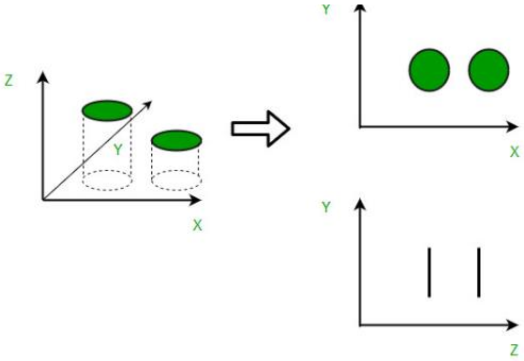


Figure III.5. Example of dimensionality reduction [61]

3.3.1 Dimension reduction techniques

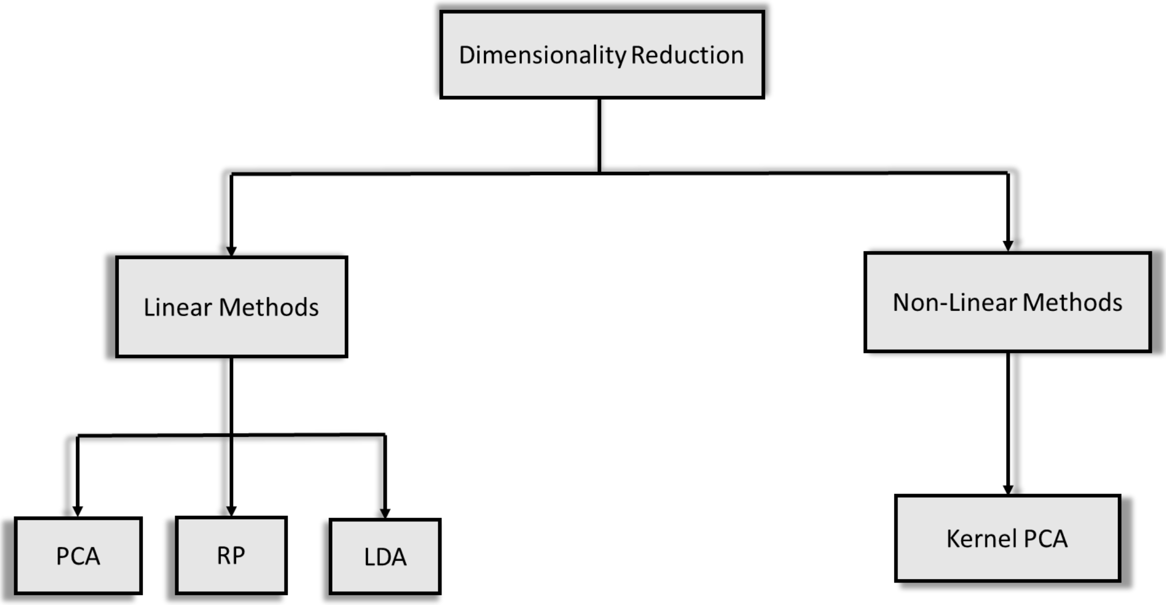


Figure III.6. Dimension Reduction Techniques

Dimension reduction techniques are employed to reduce the number of features in a dataset while preserving the essential information. These techniques aim to overcome the challenges posed by high-dimensional data, such as computational complexity, overfitting, and the curse of dimensionality. Dimension reduction techniques are grouped in 2 categories which are: linear and nonlinear techniques.

- **Nonlinear techniques:** When the data is not linearly separable, Non-linear dimension reduction techniques are used to capture the non-linear relationships and structures present in the data. There are many different non-linear dimension reduction techniques available, but some of the most popular techniques include:
    - Kernel Principal Component Analysis (KPCA) which extends the traditional PCA by using kernel functions to map the data into a higher-dimensional feature space. It then performs PCA in this feature space, allowing for non-linear dimension reduction.
    - Isomap which works by finding a low-dimensional manifold that preserves the geodesic distances between points in the high-dimensional space.
    - t-Distributed Stochastic Neighbor Embedding (t-SNE) that is often used for visualization purposes. It works by minimizing the Kullback-Leibler divergence between the high-dimensional and low-dimensional distributions of the data.

Non-linear dimension reduction techniques are a powerful tool that can be used to deal with nonlinear data. However, it is important to note that these techniques are not always necessary. If the data is linearly separable, then linear dimension reduction techniques can be used to achieve the same results.

- **Linear techniques:** linear dimensionality reduction methods have been developed and have become essential tools for analyzing high-dimensional and noisy data. These methods aim to create a lower-dimensional representation of the original data while preserving important features. Linear dimensionality reduction techniques have various applications, including data visualization, exploration of data structure, data denoising, data compression, and extraction of meaningful feature spaces. There are several popular linear dimensionality reduction techniques that are widely used in various domains. Some of the commonly employed methods include:
    - **Principal Component Analysis (PCA)**.
    - **Linear Discriminant Analysis (LDA)**.
    - Independent Component Analysis (ICA).
    - **Random projection (RP)**.
    - Canonical Correlation Analysis (CCA).

These are just a few examples of linear dimensionality reduction techniques, and there are many more variations and extensions available. The choice of technique depends on the specific characteristics of the data and the objectives of the analysis.

3.3.1.1 Principal Component Analysis

Principal Component Analysis (PCA) is an unsupervised method that involves performing a computation known as Eigenvalue decomposition on a data covariance matrix or singular value decomposition on a data matrix. Typically, the data is first centered by subtracting the mean for each attribute. The outcomes of PCA are commonly discussed in terms of component scores and loadings. It is considered the simplest eigenvector-based multivariate analysis technique. Essentially, PCA reveals the internal structure of data in a manner that best explains the variance within the data. If a multivariate dataset is visualized as coordinates in a high-dimensional space, PCA provides a lower-dimensional representation, akin to a "shadow," that highlights the most informative perspective.

PCA shares a close relationship with factor analysis, to the extent that some statistical software packages blur the distinction between the two techniques. However, factor analysis operates under different assumptions regarding the underlying structure and solves eigenvectors of a slightly modified matrix. Mathematically, PCA is defined as an orthogonal linear transformation that reorients the data into a new coordinate system. This transformation ensures that the greatest variance, as projected onto the data, aligns with the first coordinate, the second greatest variance with the second coordinate, and so forth. PCA is theoretically the optimal transformation for a given dataset in terms of least square errors.

In the case of a data matrix, XT, with a zero empirical mean (obtained by subtracting the empirical mean of the distribution from the dataset), each row represents a distinct experiment repetition, while each column corresponds to the results from a particular probe.

Given a set of points in Euclidean space, the first principal component (the eigenvector with the largest eigenvalue) represents a line passing through the mean that minimizes the sum squared error with those points. The second principal component is derived from the same concept but with the removal of all correlation with the first principal component from the points. Each eigenvalue indicates the amount of variance associated with its respective eigenvector. Consequently, the sum of all eigenvalues equals the sum of squared distances between the points and their mean divided by the number of dimensions.

Essentially, PCA rotates the set of points around their mean to align them with the first few principal components. This alignment aims to maximize the preservation of variance by utilizing a linear transformation in the first few dimensions. The values in the remaining dimensions become highly correlated, allowing them to be discarded with minimal loss of information [62].

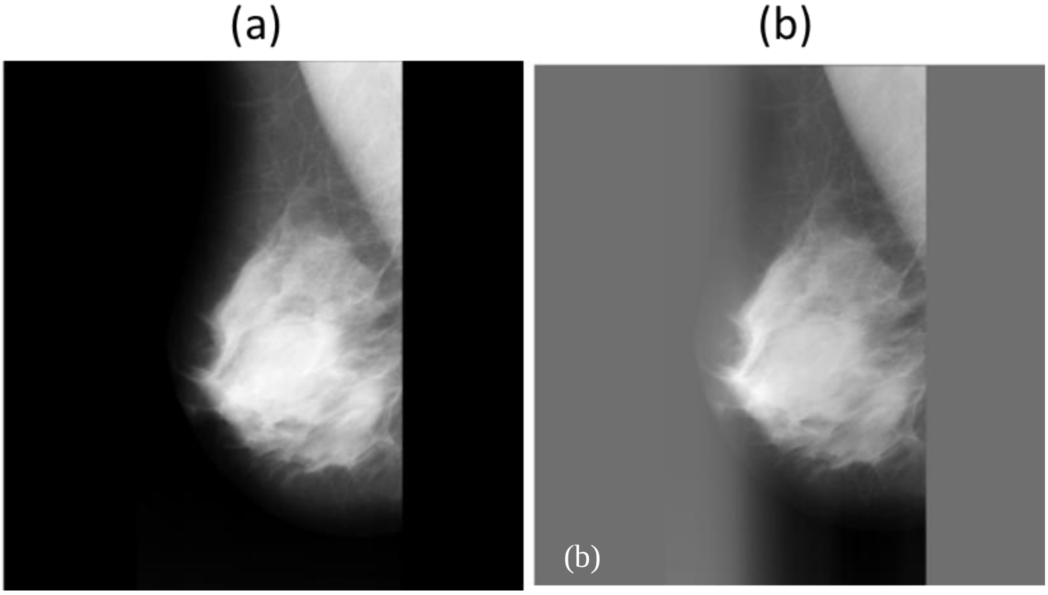


Figure III.7. mdb001 (a) Original image (b) Transformed image using PCA with 400 components

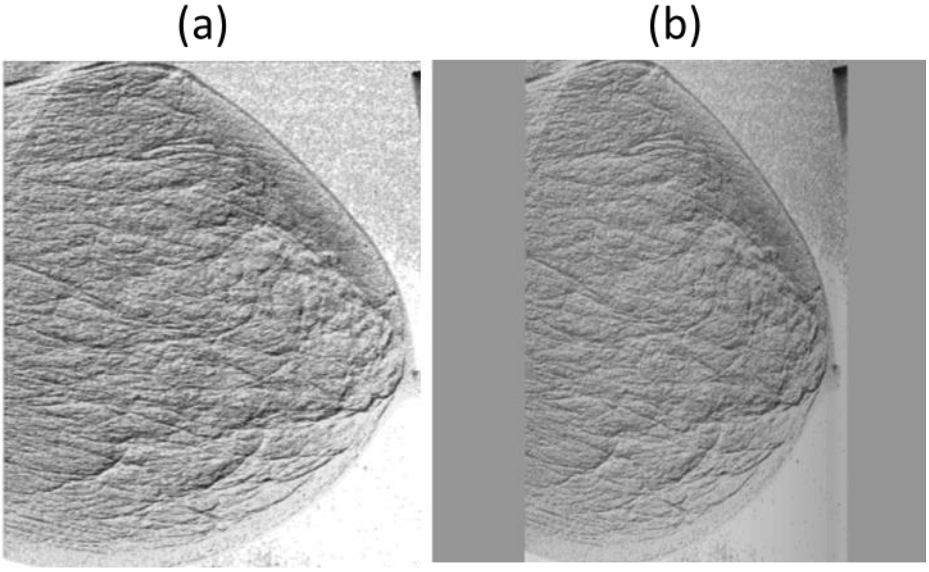


Figure III.8. mdb050 with LBP (a) Original image (b) Transformed images using PCA with 800 components

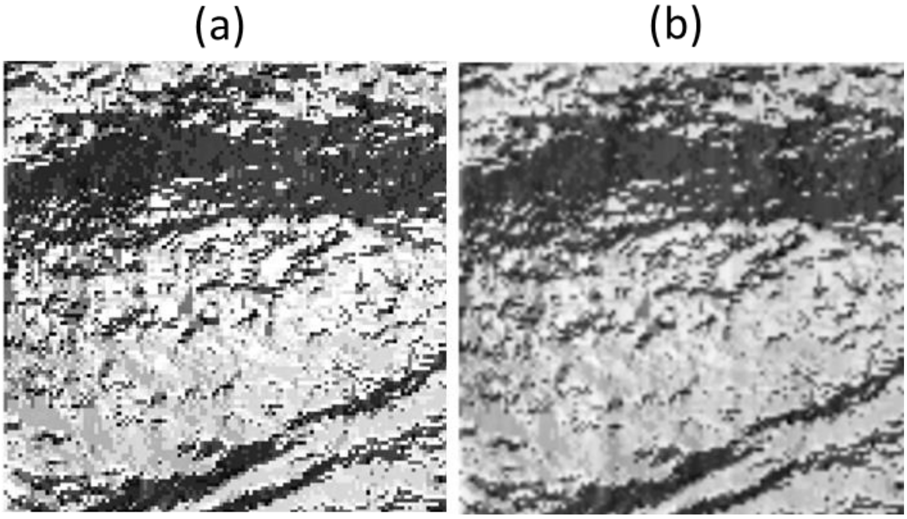


Figure III.9. mdb026 extracted ROI with LBP (a) Original image (b) Transformed images using PCA with 200 components

3.3.1.1.1 Steps of PCA:

There are several steps involved in PCA.

1. **Standardization:** The objective of this step is to normalize the range of the continuous initial variables, ensuring that each variable contributes equally to the analysis. This standardization process ensures that no single variable dominates the analysis due to its larger or smaller range of values. This can be done by following this formula:

$$z = \frac{x - \mu}{\sigma} \tag{6}$$

Where x is the value of the sample, μ is the mean of the feature and σ is the standard deviation.

2. **Covariance Matrix Computation:** The purpose of this step is to analyze the relationship between variables in the input dataset and observe how they deviate from the mean in relation to each other. This helps us determine if there are any correlations among the variables, as sometimes variables can be highly correlated and contain redundant information. To identify these correlations, we calculate the covariance matrix.

$$COV(X,Y) = \frac{\sum_{i=1}^{n} (X - \mu(X)) * (Y - \mu(Y))}{n} \tag{7}$$

n is the number of samples and X; Y are the two features of the dataset.

3. **Eigenvectors and Eigenvalues / Singular Value Decomposition (SVD) computation:** In order to determine the principal components, eigenvectors and eigenvalues must be calculated. The eigenvalues can be determined by solving this equation:

$$\det(A - \lambda I) = 0 \tag{8}$$

A is the covariance matrix, λ is the eigenvalue and I is the identity matrix. After finding the eigenvalues we have to solve this equation to get the eigenvectors associated to these eigenvalues:

$$Av = \lambda v \tag{9}$$

v represents the eigenvectors or in other words the principal components which have to be ranked in descending order. To compute the percentage of variance (information) accounted for by each component, we divide the eigenvalue of each component by the sum of eigenvalues.

4. **Feature Vector:** In this step, what we do is, to choose whether to keep all these components or discard those of lesser significance (of low eigenvalues), and form with the remaining ones a matrix of vectors that we call *Feature vector*. This is the first step towards dimension reduction because we keep p features from n features with p<<n
5. **Recast the data along the Principal Component Axis:** The purpose is to utilize the feature vector created from the eigenvectors of the covariance matrix to transform the data from its original axes to the axes represented by the principal components, giving rise to the term Principal Components Analysis. This transformation can be achieved by multiplying the transpose of the original dataset by the transpose of the feature vector [63].

$$Final\ Dataset = FeatureVector^{T} * StandardizedOriginalDataset^{T} \tag{10}$$

3.3.1.1.2 Advantages and limitations of PCA

Table III.2. PCA advantages and limitations

Advantages	Limitations
Easy to understand and implement	Information loss
Effective in reducing dimensionality	Assumes linearity
Prevents overfitting	Interpretation of components
Removes correlated features	Computational Complexity

3.3.1.2 Random Projection

Data projections involve the use of a transformation matrix, denoted as D_o * D_p, where D_o represents the dimension of the original dataset space, and Dp represents the dimension of the projected space. When an instance x is projected, its value vector is multiplied by this matrix to obtain a new vector, referred to as x projection. Projections are commonly employed to reduce the dimensionality of a dataset (i.e., Do > Dp) with the goal of noise reduction or computational speedup. Most projection methods restrict the size of the resulting transformation matrix to be smaller than or equal to. However, in the case of Random Projections (RPs), it is possible to obtain a projected space that is larger than the original space since the matrix entries are simply random numbers [64]. Random projection is an unsupervised method which involves the projection of the original d-dimensional data onto a lower-dimensional subspace with k dimensions, where k is significantly smaller than d. This projection is achieved by utilizing a random matrix R with dimensions k * d, where each column of R has unit length. In matrix notation, if X_{d*n} represents the original set of n d-dimensional observations, then x_{k*n}^{Rp} = Rk ∗ d ∗ Xd ∗ n denotes the projection of the data onto the lower k-dimensional subspace [65].

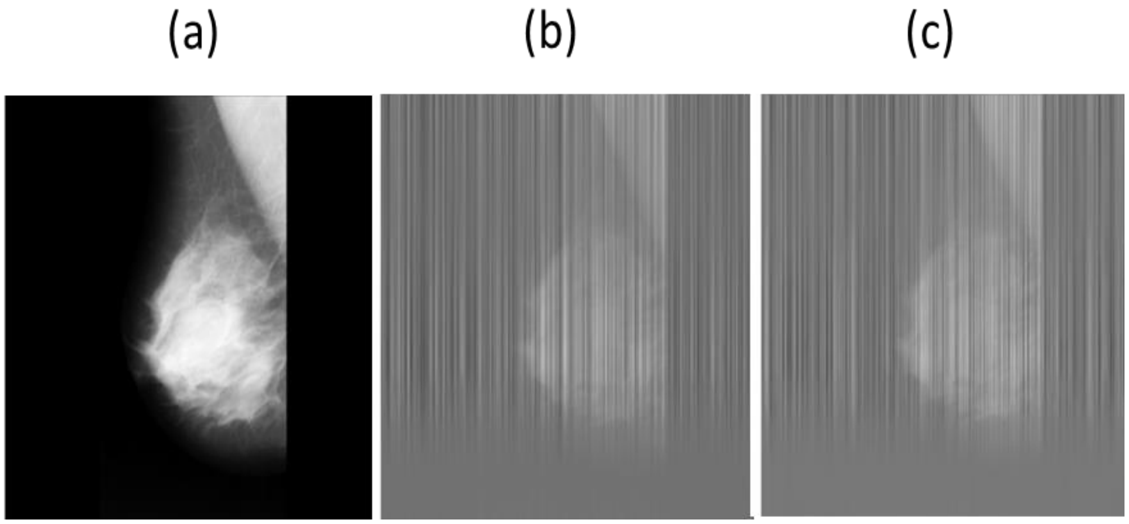


Figure III.10. mdb001 (a) Original image (b) Transformed image using GRP with n_components=400 (c) Transformed image using SRP with n_components=400

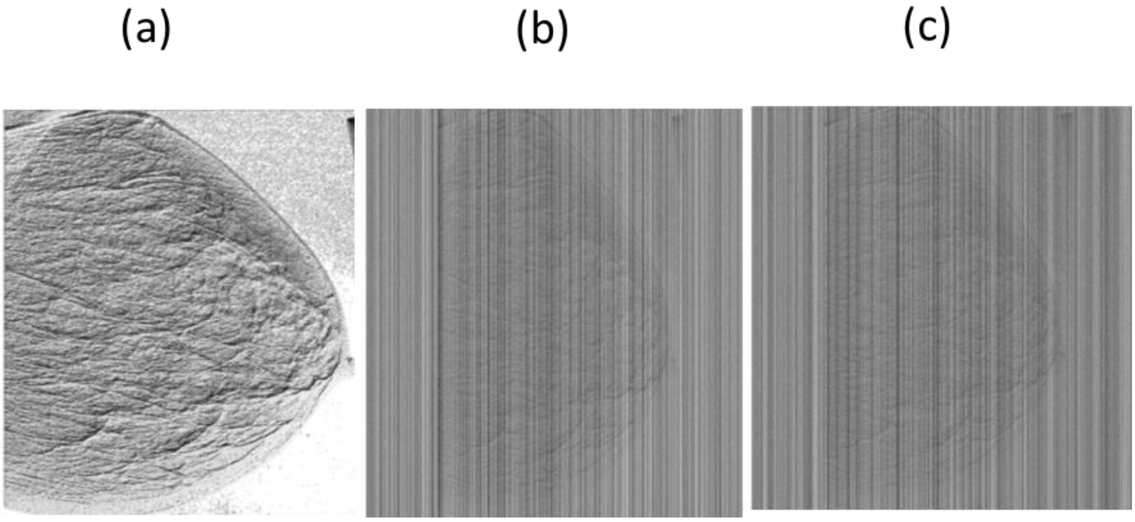


Figure III.11. mdb050 with LBP (a) Original image (b) Transformed image using GRP with n_components=800 (c) Transformed image using SRP with n_components=800

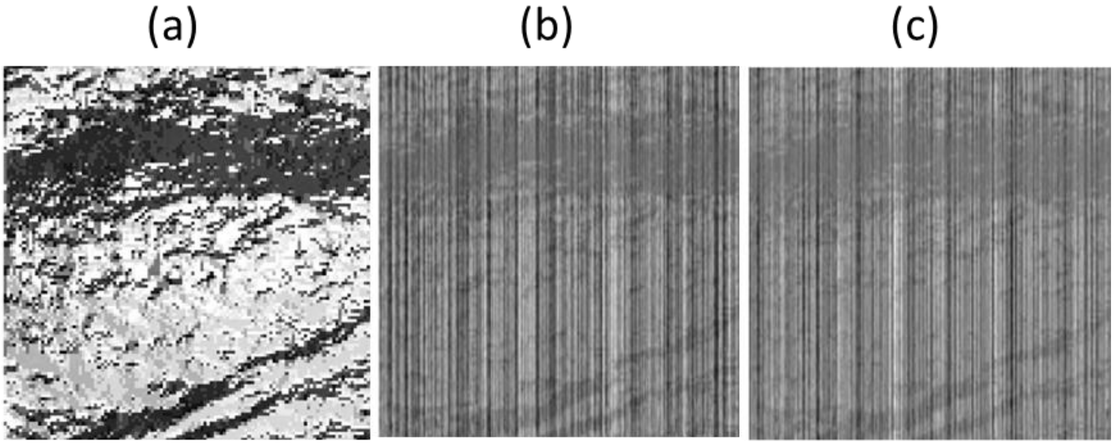


Figure III.12. mdb026 extracted ROI with LBP (a) Original image (b) Transformed image using GRP with n_components=800 (c) Transformed image using SRP with n_components=800

3.3.1.2.1 Projection types

Random projection can be performed using different types of projection matrices, such as Johnson-Lindenstrauss (JL), Gaussian, and sparse projections.

○ **Johnson Lindenstrauss Lemma RP:** This method ensures better preservation of distances between points in the original space. It is based on the Johnson and Lindenstrauss theorem. This theorem states that for a given ε > 0, an integer n which is the original dimension, and a positive integer k which represents the new dimension such that $k \geq k_0 = O(\varepsilon^{-2}*\log(n))$, for every set P of n points in R_d, there exists a function f: $R_d \rightarrow R_k$ such that for all u, v ∈ P:

$$(1-\varepsilon)\left||u-v|\right|^2 \leq \left||f(u)-f(v)|\right|^2 \leq (1+\varepsilon)\left||u-v|\right|^2 \qquad (11)$$

This theorem can be used to determine the value of the new dimension d following some formulae as sited in some papers. Some of them are:

$$\mathrm{D} > \left[9 \, \frac{1}{\varepsilon^2 - \frac{2}{3}\varepsilon^3} \, \log(\mathrm{M})\right], \tag{12}$$

Or

$$\mathrm{D} > \left[4 \, \frac{1}{\frac{\varepsilon^2}{2} - \frac{\varepsilon^3}{3}} \, \log(\mathrm{M})\right] \tag{13}$$

The other methods aim to preserve pairwise Euclidean distances in the projected space, which aligns with the requirements of the Johnson-Lindenstrauss lemma.

○ **Gaussian Random Projection (GRP):** Each entry in the transformation matrix is drawn from a Gaussian distribution with a mean μ=0 and a standard deviation σ=1/d.

○ **Sparse Random Projection (SRP):** Sparse random matrices offer an alternative to dense random projection matrices, providing comparable embedding quality while exhibiting significantly higher memory efficiency and enabling faster computation of the projected data [66]. Noting s=1/density the components of the SRP are drawn from:

$$R_{\mathrm{ij}} = \begin{cases} +\dfrac{sqrt(s)}{sqrt(n_{components})} & with\ probability\ of\ \ 1/2s \\ \quad 0 & with\ probability\ of\ \ 1-1/s \\ -\dfrac{sqrt(s)}{sqrt(n_{components})} & with\ probability\ of\ \ 1/2s \end{cases} \tag{14}$$

Where the density=sqrt (original dimension) which is a value in]0,1] and $n_{components}$ is the new dimension. Achlioptas [65] has recently shown that the Sparse distribution can be replaced by a much simpler distribution by replacing the density with 1/3 as follows:

$$R_{\mathrm{ij}} = \sqrt{3}\begin{cases} +1\ with\ probability\ of\ \ 1/6 \\ \ \ 0\ with\ probability\ of\ \ 2/3 \\ -1\ with\ probability\ of\ 1/6 \end{cases} \tag{15}$$

Achlioptas's finding leads to additional computational savings in database applications by enabling the use of integer arithmetic for the computations. The Sparse projection is expected to contribute to diversity due to the presence of zeros. These zeros effectively exclude certain original dimensions from the computation of some of the new dimensions. There are successful ensemble methods that also train their base classifiers by excluding certain features.

3.3.1.2.2 Advantages and limitations of RP

Table III.3. Random projection advantages and limitations

Advantages	Limitations
Preserve Distances	Information loss
Computational Efficiency	Non-Linear Data Mapping
Simple and efficient	Sensitivity to Projection Parameters
Data visualization	Sensitive to noise

3.3.1.3 Linear Discriminant Analysis

Linear Discriminant Analysis (LDA) is a widely used approach for both classification and dimensionality reduction. LDA effectively handles scenarios where the frequencies within classes are unequal, and its performance has been evaluated using randomly generated test data. By maximizing the ratio of between-class variance to within-class variance in a given dataset, LDA ensures optimal separability. Despite its simplicity, LDA often yields robust, satisfactory, and easily interpretable classification outcomes. In real-world classification problems, LDA is frequently employed as a first and benchmarking method before utilizing more complex and flexible techniques [67].

3.3.1.3.1 LDA steps

Here's a step-by-step explanation of how LDA works:

- **Mean vectors computation:** The mean of each class has to be calculated using this formula:

$$\mu_i = \frac{1}{n_i} \sum_{x \in D_i}^{n} x_k \tag{16}$$

Where ni represents the number of samples and x_k is the value of the sample.

- **Scatter matrices computation:** In order to calculate the scatter matrix, The within-class and the between-class scatter matrix have to be calculated.
    - **Within-class scatter matrix (Sw):** The within-class scatter matrix measures the spread or variance of the data within each class. It provides information about the distribution of the data points within their respective classes. The Sw is calculated using the following equation:

$$\mathrm{Sw} = \sum_{\mathrm{i}=1}^{\mathrm{c}} \mathrm{Si} \tag{17}$$

Where

$$\mathrm{Si} = \sum_{\mathrm{x} \in \mathrm{Di}}^{\mathrm{c}} (\mathrm{x} - \mu\mathrm{i})(\mathrm{x} - \mu\mathrm{i})^{\mathrm{T}} \tag{18}$$

Si is the scatter matrix of each class and μ*i* is the mean vector (same computation formula as in PCA).

- **Between-class scatter matrix (Sb):** The between-class scatter matrix captures the variation between different classes and provides information about the separability of the classes. It is computed as the weighted sum of the covariance matrices of the class means. It's computed using the following formula:

$$Sb = \sum_{i=1}^{c} ni\,(\mu i - \mu)(\mu i - \mu)^{T} \tag{19}$$

Where μ is the overall mean, μi and ni are the sample mean and sizes of the respective classes.

- **Eigenvalues and eigenvectors computation:** The computation of eigenvalues and eigenvectors involves finding the directions that maximize the separability between classes. To compute the eigenvectors, the following equation is used:

$$\det\,(Sw^{-1}Sb - \lambda I) = 0 \tag{20}$$

Where

$$A^{-1} = (\mathrm{adj}\ (\mathrm{A}))/(\det\ (\mathrm{A})) \tag{21}$$

Adj(A)= Transpose of Cofactor Matrix (C^{T}) calculated by this equation:

$$Cij = ((-1)^{i+j})Mij \tag{22}$$

M is the minor matrix.

Once the eigenvalues (λ) are obtained we can calculate their eigenvectors by solving this equation:

$$(Sw^{-1}Sb - \lambda I)\,W = 0 \tag{23}$$

Where W is the eigenvector(s) which are then sorted in descending order and arranged their corresponding eigenvectors accordingly to choose the ones to keep (most informative).

Elementary Row Operations can also be used to calculate the inverse matrix.

- **Transforming the samples into a new subspace:** The kept eigenvectors are used to recast the data into a new dimension by multiplying them with the original data (projection to a new dimension) using this equation:

$$Y = X * W \tag{24}$$

Where X is an $n \times d$-dimensional matrix representing the n samples, and Y is the transformed $n \times k$-dimensional samples in the new subspace.

3.3.1.3.2 Advantages and limitations of LDA

Table III.4. LDA advantages and limitations

Advantages	Limitations
Class separability	Curse of dimensionality (features>>samples)
Simple and interpretable	Independence assumption
Multi-class classification	Limitations in high-dimensional spaces
Less prone to overfitting	Linearity assumption

In summary, each dimension reduction technique has its own strengths, limitations, and assumptions. The choice of technique depends on the specific characteristics of the dataset, the desired preservation of information, and the downstream analysis tasks. Dimension reduction techniques play a crucial role in preprocessing and analyzing high-dimensional data by simplifying the data representation, improving computational efficiency, and aiding in data visualization and interpretation.

3.4 Conclusion

In conclusion, dimension reduction and feature selection techniques play a crucial role in data analysis and machine learning. As datasets become larger and more complex, extracting meaningful insights becomes increasingly important. The application of dimension reduction and feature selection allows us to effectively navigate high-dimensional data, uncover hidden patterns, and construct more robust and efficient machine learning models. These techniques preserve information, reduce complexity and empower us to make informed decisions based on concise and meaningful features. In the coming chapter, we investigate the fundamental task of classification which is a key component of machine learning, enabling us to make predictions and decisions based on labeled training data

IV. Chapter 4

Classification

4.1 Introduction

The classification is a fundamental aspect of machine learning and data analysis, focusing on the assignment of predefined labels or categories to input data based on observed features. It plays a crucial role in various applications, enabling informed decision-making, outcome prediction, and data-driven insights. We explore popular classification algorithms such as K-Nearest Neighbors (KNN), Support Vector Machines (SVM). Additionally, we examine dimension reduction techniques like Principal Component Analysis (PCA), Random Projection (RP) and Linear Discriminant Analysis (LDA) to improve classification accuracy. We address challenges in classification tasks, including imbalanced datasets, missing data, and noisy features. Evaluation metrics such as accuracy, precision, recall, F1-score, sensitivity and specificity are used to assess the performance of classification models. We also discuss model selection, parameter tuning, and validation techniques to ensure reliable and robust results.

In this chapter, we delve into the principles, algorithms, and methodologies that underpin the classification process. Our objective is to understand how machines can autonomously learn from patterns in data and accurately classify new instances.

4.2 Artificial intelligence

Artificial intelligence (AI) encompasses a broad field of computer science focused on creating intelligent machines that can perform tasks typically requiring human intelligence. It combines various interdisciplinary approaches, but recent advancements in machine learning and deep learning have particularly revolutionized the technology landscape [68]. AI systems use various techniques such as **machine learning**, deep learning, neural networks, and natural language processing to recognize patterns, process data, and make predictions. In general, AI systems work by ingesting large amounts of labeled training data, analyzing the data for correlations and patterns, and using these patterns to make predictions about future states [69]. AI programming focuses on cognitive skills that include the following:

- **Learning:** This aspect of AI programming focuses on acquiring data and creating rules for how to turn it into actionable information.
- **Reasoning:** This aspect of AI programming focuses on choosing the right algorithm to reach a desired outcome.
- **Self-correction:** This aspect of AI programming is designed to continually fine-tune algorithms and ensure they provide the most accurate results possible.

- **Creativity:** This aspect of AI uses neural networks, rules-based systems, statistical methods and other AI techniques to generate new images, new text, new music and new ideas.

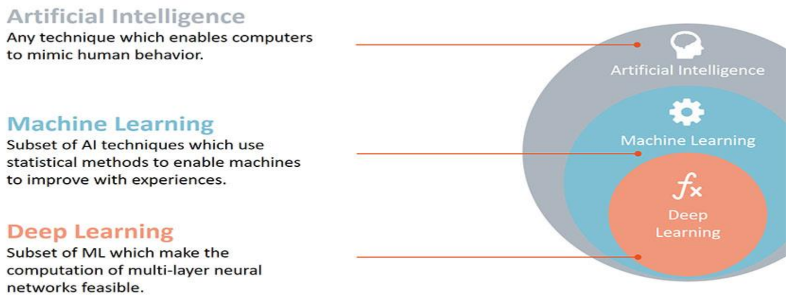

Artificial Intelligence
Any technique which enables computers to mimic human behavior.
Machine Learning
Subset of AI techniques which use statistical methods to enable machines to improve with experiences.
Deep Learning
Subset of ML which make the computation of multi-layer neural networks feasible.
Artificial Intelligence
Machine Learning
fx
Deep Learning

Figure IV.1. Artificial intelligence, machine learning and deep learning [70]

4.2.1 Types of AI

Experts have divided the AI field into weak AI and strong AI:

- **Weak AI:** Weak AI also known as narrow AI, is designed and trained to complete a specific task. Industrial robots and virtual personal assistants, such as Apple's Siri, use weak AI [69].
- **Strong AI:** also known as artificial general intelligence (AGI), describes programming that can replicate the cognitive abilities of the human brain [69]. These tend to be more complex and complicated systems. They are programmed to handle situations in which they may be required to problem solve without having a person intervene. These kinds of systems can be found in applications like self-driving cars or in hospital [71].

4.2.2 AI applications

There are various, real-world applications of AI systems today. Some of the most common use cases are:

- **Speech recognition: by** processing human speech into a written format.
- ***Computer vision:*** **by** extracting meaningful information from digital images, videos and other visual inputs, and from these inputs, it can take action.
- **Education:** by automating grading, giving educators more time for other tasks and assessing students and adapting to their needs, helping them work at their own pace.
- ***Healthcare:*** by suggesting drug dosages, identifying treatments, and for aiding in surgical procedures in the operating room.
- **Finance:** by optimizing stock portfolios, AI-driven high-frequency trading platforms making thousands or even millions of trades per day without human intervention.

4.2.3 Advantages and limitations of AI

Table IV.1. Advantages and disadvantages of AI

Advantages	Limitations
Reduces human error	High costs
24/7 support	Unemployment
Perform repetitive jobs	Makes humans lazy
Makes faster decisions	Has no ethics
Digital assistance	Lack of emotions
Medical applications	Dependency on machines

4.3 Machine Learning

Machine learning (ML) is a field of study within artificial intelligence and computer science, involves using data and algorithms to teach computers how to learn and improve their performance over time. It mimics the way humans learn, gradually enhancing its accuracy and capabilities [72]. In simple technical terms, ML employs algorithms to process empirical or historical data, enabling the generation of outputs through analysis. In certain methodologies, these algorithms initially operate on "training data" to learn, predict, and continuously enhance their performance over time [73].

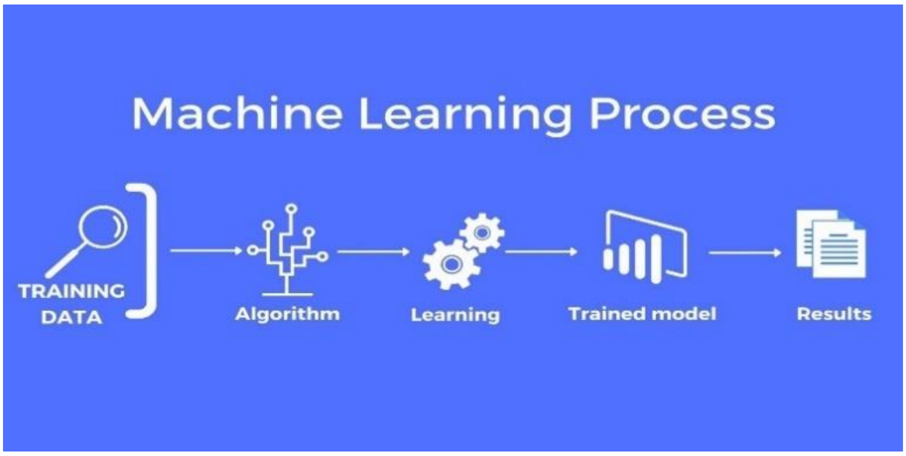


Figure IV.2. Machine learning process [74]

4.3.1 Deep learning

Deep learning is a specific branch of machine learning that harnesses the power of neural networks to comprehend intricate patterns within data. These neural networks consist of interconnected nodes, resembling the structure of neurons in the human brain. Through training, these networks can discern and understand patterns in data, enabling them to generate accurate predictions based on the learned patterns. Deep learning has achieved remarkable success in various domains, particularly in tasks such as image recognition and natural language processing. As the forefront of artificial intelligence, deep learning is extensively employed across diverse applications, including self-driving cars, voice recognition systems, and cancer diagnosis. Its ability to uncover complex relationships within data has made it a driving force in advancing technological solutions.

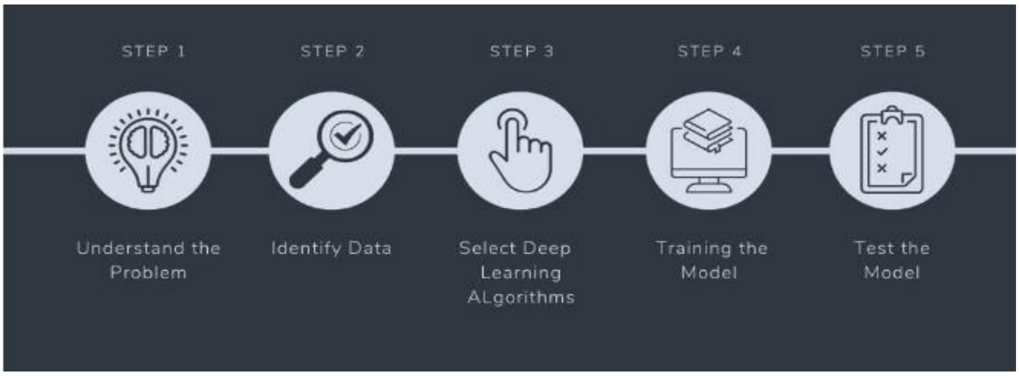


Figure IV.3. Deep learning process [75]

4.3.2 Neural network

Neural networks, also known as artificial neural networks (ANNs) or simulated neural networks (SNNs), are a subset of machine learning algorithms inspired by the human brain. They consist of interconnected nodes that mimic the behavior of biological neurons. These networks are organized into layers, including an input layer, one or more hidden layers, and an output layer. Nodes in each layer have associated weights and thresholds, determining their activation. Through training with data, neural networks learn to recognize patterns and make predictions. Neural networks have proven highly effective in various domains, such as image recognition and natural language processing. They have become a cornerstone of deep learning, driving advancements in artificial intelligence. Applications of neural networks include self-driving cars, voice recognition systems, and medical diagnoses. Once trained, neural networks can rapidly process and classify data, outperforming human experts in tasks like speech and image recognition. Notably, Google's search algorithm employs neural networks for efficient information retrieval [76].

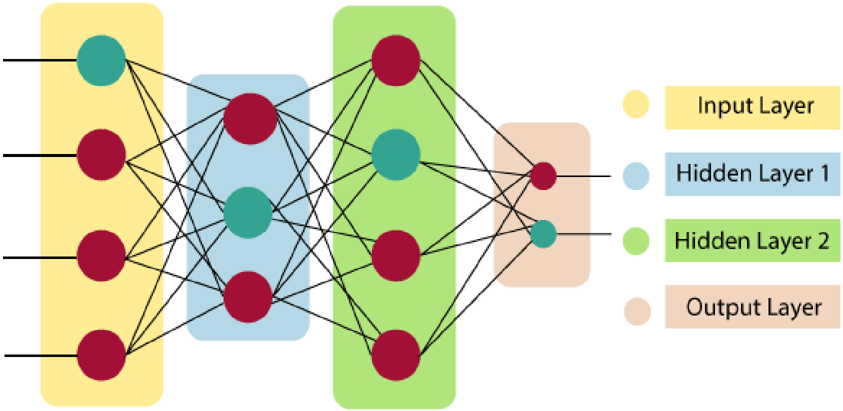


Figure IV.4. Artificial neural network architecture [77]

4.3.3 Machine learning technology

Machine learning technology includes a range of methodologies and tools employed to construct machine learning models. This technology involves several essential components, including feature engineering which involves the creation of meaningful features or variables that enable machine learning algorithms to learn patterns and make accurate predictions. It entails transforming raw data into informative representations that capture relevant information for the task at hand, data preprocessing which plays a pivotal role in preparing data for utilization by machine learning algorithms. It encompasses various operations, such as data cleaning to handle missing values or outliers, data scaling to ensure features are on comparable scales, and data normalization to enhance model performance, and model selection that involves the careful consideration and choice of an appropriate machine learning algorithm for a specific problem. This process entails evaluating various algorithms, taking into account factors such as their performance metrics, computational requirements, and suitability for the given task. By employing feature engineering, data preprocessing, and model selection techniques, machine learning practitioners can enhance the performance and reliability of their models for a wide range of applications.

4.3.4 Machine learning model

Machine learning models come into existence through the application of machine learning algorithms to data. These models can be classified into two main categories: linear and nonlinear models, each with its own characteristics.

Linear models are known for their simplicity and ease of interpretation. They offer straightforward insights into the relationship between input variables and the predicted outcome. However, linear models often struggle to handle complex problems, and their performance may be limited in such scenarios.

On the other hand, nonlinear models exhibit greater flexibility and can capture intricate relationships within data. They excel at tackling complex problems by incorporating more sophisticated patterns and interactions. However, due to their increased complexity, nonlinear models may require more advanced techniques for training and interpretation.

The choice between these two types of models depends on the nature of the problem at hand, striking a balance between interpretability and predictive power.

4.3.5 Machine learning types

Machine learning encompasses various types including **supervised learning**, **unsupervised learning**, reinforcement learning, semi-supervised learning, self-supervised learning and transfer learning.

- **Supervised learning:** Supervised machine learning involves training models with labeled data sets, enabling them to learn and improve their accuracy progressively. For instance, an algorithm can be trained with a collection of labeled images, including pictures of dogs and other objects, empowering the machine to autonomously identify dog pictures. This form of machine learning, known as supervised learning, is widely prevalent in present-day applications [78]. As a part of supervised machine learning, classification has achieved speculations rise.
- **Unsupervised learning:** Unsupervised machine learning involves algorithms that search for patterns in unlabeled data, uncovering hidden insights and trends that may not be apparent to human observers. For instance, an unsupervised machine learning program can analyze online sales data and identify distinct clusters of customers based on their purchasing behavior, without any prior knowledge or explicit guidance. This approach allows for the

discovery of valuable patterns and relationships that can inform decision-making and provide valuable insights [78].

- **Reinforcement learning:** Reinforcement machine learning involves training machines through a trial-and-error process to make optimal decisions based on a reward system. This approach is commonly used to teach models to play games or train autonomous vehicles to navigate. By providing feedback to the machine when it makes correct decisions, reinforcement learning enables the machine to learn and refine its actions over time. This iterative process allows the model to gradually develop the knowledge and strategies necessary to make informed choices in various scenarios [78].
- **Semi-supervised learning:** Semi-supervised learning is a hybrid technique between supervised and unsupervised learning in which the core idea is to treat data points differently based on their label availability. For labeled points, the algorithm leverages traditional supervision techniques to update the model weights. In contrast, for unlabeled points, the algorithm focuses on minimizing the prediction disparities with other similar training examples [79].
- **Self-supervised learning:** Self-supervised learning is a specialized branch within the realm of unsupervised learning that capitalizes on the utilization of unlabeled data. The primary concept revolves around enabling the model to learn data representations autonomously, without relying on manual labels. By acquiring the ability to comprehend and capture meaningful patterns within the data, the model can subsequently be employed for downstream tasks, even with a reduced amount of labeled data, while achieving comparable or improved performance compared to models trained without self-supervised learning [80]. For example, a self-supervised learning model could be trained to predict the next word in a sentence by being given a dataset of sentences that have been truncated after the first few words.
- **Transfer learning:** Transfer learning is a technique in machine learning that involves leveraging a previously developed model as the foundation for a new model targeting a different task. This method is particularly prevalent in the field of deep learning, where pre-existing models are employed as starting points for computer vision and natural language processing tasks. The adoption of pre-trained models is favored due to the substantial computational and time resources required to build neural network models for such problems. Moreover, pre-trained models offer significant performance improvements when applied to related problems, making them highly advantageous in various domains [81].

4.3.6 Classification

Classification in machine learning involves predicting a class label for a given input data instance. To achieve this, a classification model is trained using a dataset comprising numerous input-output pairs. The model learns the optimal mapping between input data examples and their corresponding class labels by analyzing the training dataset.

For effective classification, the training dataset must be representative of the problem at hand and contain an ample number of examples for each class label. This ensures that the model can accurately generalize and make accurate predictions for unseen data instances. By leveraging the training dataset, the classification model determines the most suitable decision boundaries or patterns to distinguish between different classes, enabling it to classify new input data correctly [82].

There are perhaps four main types of classification tasks that you may encounter; they are:

- ***Binary classification*** which is a fundamental process or task in machine learning that involves categorizing data into two distinct classes. It serves as a predictive technique aimed at determining the membership of a given data point in one of two predefined groups. By employing various algorithms and models, classification facilitates making informed predictions about the class to which a particular entity or observation belongs to [83].
- **Multi-class classification** which encompasses classification tasks that involve more than two class labels. It is employed in various applications such as face classification, plant species classification, and optical character recognition. Unlike binary classification, where outcomes are categorized as normal or abnormal, multi-class classification involves classifying examples into multiple known classes [82].
- **Multi-label classification** refers to classification tasks that involve assigning two or more class labels to each example, allowing for the prediction of multiple labels for a single instance. Unlike binary and multi-class classification, where a single class label is assigned to each example, multi-label classification enables the assignment of multiple labels simultaneously. To handle multi-label classification tasks, it is common to use models that generate multiple outputs. These outputs are typically treated as independent binary classification predictions, where each output represents the probability of the corresponding label being present in the example. This approach

allows for the modeling of complex relationships between the input data and multiple labels [82].

- **Imbalanced classification** refers to classification tasks in which the distribution of examples among different classes is highly uneven. In most cases, imbalanced classification involves binary classification problems, where the training dataset is characterized by a significant disparity in the number of examples between the majority class (typically the "normal" class) and the minority class (often the "abnormal" class) [82].

4.3.7 Algorithms of machine learning

There are so many different ML algorithms, each one with its own weaknesses and strengths. Some of these algorithms are:

- Linear regression
- Random forest
- Logistic regression
- Decision trees
- **Support vector machines (SVMs)**
- **K-nearest neighbors (KNN)**

The choice of the algorithm to use for a particular problem depends on the specific data and the desired outcome. Here are some of the factors that can help for choosing the best one:

- The problem type, for example linear or non-linear.
- The quantity of data available.
- Speed and accuracy requirements.
- The available resources.

Machine learning is an influential tool with broad applications for problem-solving. However, it is crucial to acknowledge that machine learning algorithms are not perfect. They are susceptible to errors and biases. It is essential to critically assess the outcomes of machine learning algorithms and employ them alongside other approaches to ensure the attainment of precise and reliable results.

4.3.7.1 K-Nearest Neighbors

The k-nearest neighbors algorithm, commonly referred to as KNN or k-NN, is a supervised learning classifier that falls under the non-parametric category. It utilizes the concept of proximity to make predictions or classifications concerning the grouping of individual data points. While it can be employed for both regression and classification tasks, KNN is primarily utilized as a classification algorithm. It operates on the premise that similar data points tend to cluster in close proximity to each other [48]. Additionally, it is important to mention that the KNN algorithm belongs to the category of "lazy learning" models, which means that it does not learn a model from the training data. Instead, it simply stores the training data and then uses it to make predictions when new data points are presented. This makes KNN a very fast algorithm, but it can also make it less accurate than other supervised learning algorithms that learn a model from the training data. In other words, all training samples are stored in a pattern space with multiple dimensions. The classification of an unknown sample is determined by taking a majority vote from its neighboring samples within the training pattern space [84]. It depends on 2 essential parameters which are the **distance** metric used and the value of **K** that are determined by the user. The process of KNN is explained by the figure shown below.

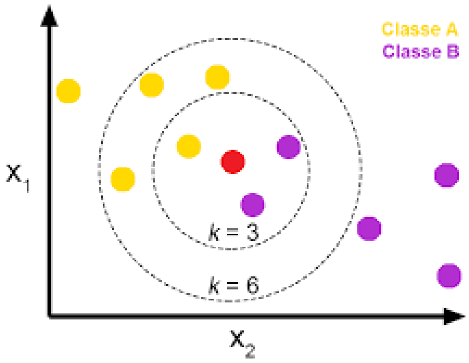


Figure IV.5. Example of how KNN works [85]

4.3.7.1.1 KNN parameters

4.3.7.1.1.1 Distance metrics

To identify the data points nearest to a specific query point, it is necessary to compute the distances between the query point and other data points. These distance metrics play a crucial role in defining decision boundaries that separate query points into distinct regions [86]. There are many distance measures that can be used in KNN classification but we'll focus on the following metrics which are: Euclidian distance, Manhattan distance and Minkowski distance.

4.3.7.1.1.1.1 Euclidian distance (p=2)

The Euclidean distance, also known as the Cartesian distance, measures the separation between two points in a plane or hyperplane. It can be envisioned as the length of a straight line connecting the two points under consideration. This metric allows us to calculate the overall displacement between two states of an object [87].

$$d(x,y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \tag{25}$$

4.3.7.1.1.1.2 Manhattan distance (p=1)

This distance metric is commonly employed when the focus is on determining the total distance covered by an object, rather than just its displacement. This metric is computed by summing the absolute differences between the coordinates of the points across multiple dimensions [87].

$$d(x,y) = \sum_{i=1}^{n} |x_i - y_i| \tag{26}$$

4.3.7.1.1.1.3 Minkowski distance

This metric presented is a generalized version encompassing both Euclidean and Manhattan distance metrics. By introducing a parameter, denoted as p, in the following formula, it becomes possible to create various other distance metrics. When p is set to two, the formula represents the Euclidean distance, whereas a value of one for p corresponds to the Manhattan distance [86].

$$d(x,y) = \sum_{i=1}^{n} |x_i - y_i|^{1/p} \tag{27}$$

Generally, the Euclidean metric is the most commonly used distance metric.

4.3.7.1.2 K

The KNN classifier can perform efficiently, if the optimal value of 'k' is used in the classifier design. To determine the optimal value of K for your data, multiple runs of the KNN algorithm are performed using various K values. The aim is to identify the value of K that minimizes the number of encountered errors while preserving the algorithm's capability to make accurate predictions on unseen data [88]. It should be chosen based on the input data. If the input data has more outliers or noise, a higher value of k would be better. It is recommended to choose an odd value for k to avoid ties in classification [87].

4.3.7.1.3 KNN process

1. Load the data

2. Initialize K to your chosen number of neighbors

3. For each example in the data

3.1 Calculate the distance between the query example and the current example from the data.

3.2 Add the distance and the index of the example to an ordered collection

4. Sort the ordered collection of distances and indices from smallest to largest (in ascending order) by the distances

5. Pick the first K entries from the sorted collection

6. Get the labels of the selected K entries

7. If regression, return the mean of the K labels

8. If classification, return the mode of the K labels

4.3.7.1.4 Advantages and limitations

Table IV.2. KNN advantages and limitations

Advantages	Limitations
Simple and easy to implement and understand	High memory costs for big data
Adapts easily	Curse of dimensionality
Few hyperparameters	Prone to Overfitting
Versatile (can be used for classification and regression)	Sensitive to the value of k
Fast	Not as accurate as other supervised learning algorithms

In summary, KNN is a simple, versatile, and fast algorithm that can be used for a wide variety of problems such as Data preprocessing, Finance, Pattern Recognition and **images classification**. However, it is important to be aware of its limitations, such as its potential for lower accuracy and its sensitivity to the value of k.

4.3.7.2 Support Vector Machine (SVM)

The Support Vector Machine (SVM) is a popular supervised learning algorithm used for both classification and regression tasks, although it is primarily known for its effectiveness in classification. The main objective of the SVM algorithm is to determine an optimal decision boundary, known as a hyperplane, that can effectively separate data points in an n-dimensional space into distinct classes. By finding this boundary, SVM enables us to accurately classify new data points into the appropriate categories. In the process of constructing the hyperplane, SVM identifies and utilizes the extreme points or vectors that greatly influence its creation. These critical instances are referred to as support vectors, giving rise to the name Support Vector Machine for the algorithm [89]. A classification technique involves the utilization of training and testing data, which includes various data instances. Within the training set, each instance is comprised of a "target value" representing class labels and multiple "attributes" representing features [84]. The figure below shows two different categories that are classified using a decision boundary (hyperplane).

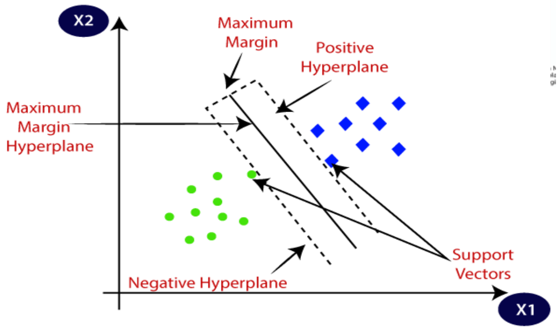


Figure IV.6. Example of SVM classifier [89]

Here are the key components and concepts related to Support Vector Machines:

- **Hyperplane:** In SVM, a hyperplane is a decision boundary that separates the data points of different classes. For binary classification, the hyperplane is a line in a two-dimensional space or a plane in a three-dimensional space. In higher dimensions, it becomes a hyperplane. The goal is to find the hyperplane that maximally separates the classes while maintaining the maximum margin.
- **Margin:** The margin is the distance between the hyperplane and the nearest data points of each class. SVM aims to maximize the margin, as a larger margin generally leads to better generalization and robustness of the model.
- **Support vectors:** SVM derives its name from the Support vectors which are the data points that lie closest to the decision boundary. These points play a crucial role in determining the optimal hyperplane.
- **Soft margin:** In cases where the data is not perfectly separable, SVM can incorporate a soft margin by allowing some misclassifications. This is achieved by introducing a slack variable that penalizes misclassifications. The trade-off between the margin and misclassification errors is controlled by the C parameter.
- **C parameter:** The C parameter in SVM controls the trade-off between maximizing the margin and minimizing the classification error on the training data. A smaller C value allows for a larger margin but may lead to misclassifications, while a larger C value reduces the margin to avoid misclassifications.

4.3.7.2.1 Types of SVM

4.3.7.2.1.1 Linear SVM

Linear Support Vector Machine (SVM) is specifically employed when dealing with linearly separable data. In other words, if a dataset can be effectively divided into two classes using a single straight line, it is referred to as linearly separable data, and the classifier used for such data is known as a Linear SVM classifier [89]. In the linear case, some e preliminaries for the support vector machine were discussed, and a straight-line equation was derived as:

$$wx_i + b = 0 \tag{28}$$

Considering a set of n data samples of two classes (x_1, y_1), (x_2, y_2)...., (x_n, y_n), i=1,2…, n, mapped to a higher dimensional space and $y_i = \pm 1$ .The separating hyper plane should be optimal for correct classification of class. Here the optimization problem is to find optimal separating hyper plane to separate positive and negative classes defined by following equations [84]:

$$(wx_i + b) \geq 1 \ \text{(For } y_i = 1) \tag{29}$$

$$(wx_i + b) \leq -1 \ \text{(For } y_i = -1) \tag{30}$$

SVM tries to maximize the margin between two classes by finding a weight vector w and bias weight b by minimizing $\frac{1}{2}\|w\|^2$ [84].

The Figure IV.7. is an example of linear SVM.

4.3.7.2.1.2 Nonlinear SVM

Non-linear Support Vector Machine (SVM) is utilized when working with non-linearly separable data. In cases where a dataset cannot be accurately classified using a straight line, it is considered non-linear data, and the classifier employed for such data is known as a Non-linear SVM classifier [89]. The classification, also referred to as domain division, can be performed in either a vector space or a feature space. In this context, the vector space represents the space that encompasses the scatter plot of the original features, while the feature space represents the space that encompasses the scatter plot of the transformed features achieved through mapping functions or kernel functions [90]. The mapping functions (also known as basis functions) are used to transform the input data from the original feature space to a higher-dimensional feature space. This transformation is performed to make the data linearly separable or to increase the separability between different classes in order to address complex classification problems that cannot be solved using linear boundaries in the original feature space.

$$\phi: R^p \rightarrow R^d$$

Rp is the vector space (original domain) and Rd is the feature space, which is high dimensional (generally d >> p). φ(x) is high dimensional, thus the computation becomes very expensive. This can be tackled using an approach called a kernel trick.

4.3.7.2.2 Kernel trick

The kernel trick is a technique used in SVM to transform the data into a higher-dimensional feature space. By using a kernel function, the algorithm can implicitly operate in this higher-dimensional space without explicitly computing the transformation. This allows SVM to handle non-linearly separable data by finding linear decision boundaries in the transformed space. Commonly used kernel functions include :

- **Linear kernel:** Computes the dot product between the original feature vectors. The linear kernel assumes that the data is linearly separable in the original feature space. It is expressed as:

$$k(x_i, x_j) = x_i^T x_j + c \tag{31}$$

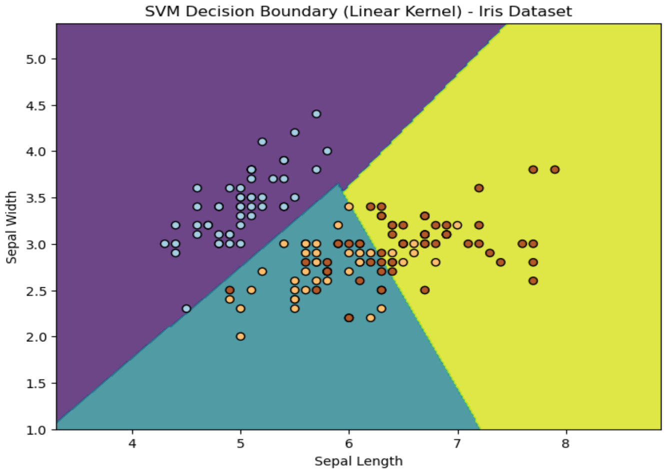


Figure IV.8. Example of SVM classification using Linear kernel

- **Polynomial kernel:** The polynomial kernel maps the original input space to a higher-dimensional feature space using polynomial functions. It allows SVM to capture non-linear relationships between the data points. The polynomial kernel is expressed as:

$$k(x_i, x_j) = (x_i^T x_j + c)^p \tag{32}$$

Putting p=2 will give a quadratic function, so the parameter p here represents the degree of the polynomial function. The figure IV.8 shows an example of the polynomial kernel.

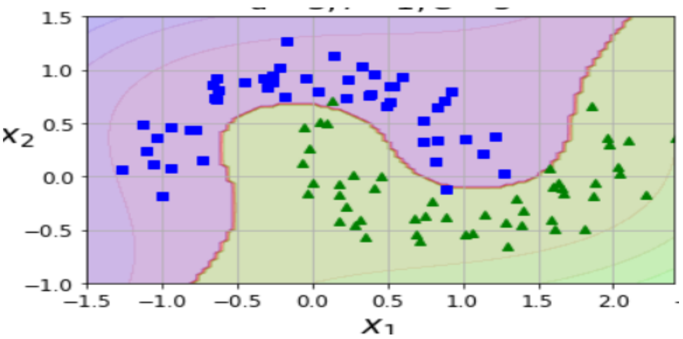


Figure IV.9. Example of SVM classification using the polynomial kernel [91]

- **Sigmoid kernel:** The sigmoid kernel applies a sigmoidal transformation to the input data, mapping it into a higher-dimensional feature space. The sigmoid kernel is expressed as:

$$k(x_i, x_j) = tanh\ (\sigma x_i^T x_j + c) \tag{33}$$

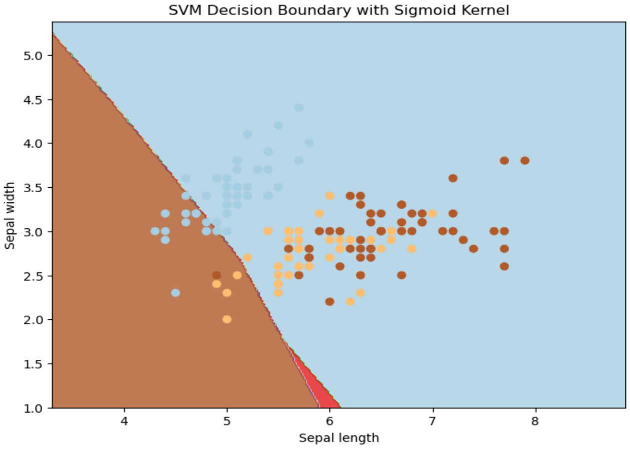


Figure IV.10. Example of SVM classification using the Sigmoid kernel

- **Radial Basis Function (RBF) Kernel:** The Gaussian kernel or the (RBF) kernel measures the similarity or distance between pairs of data points based on the radial distance from a reference point. It maps the data into an infinite-dimensional feature space. The RBF kernel is commonly used and is effective in capturing complex non-linear relationships. It is expressed as:

$$k(x_i, x_j) = e^{-\frac{\|x_i - x_j\|^2}{2\sigma^2}} \tag{34}$$

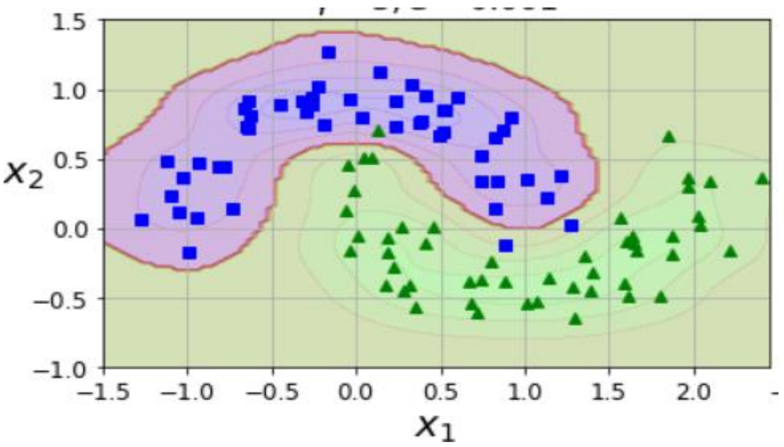


Figure IV.11. Example of SVM classification using RBF kernel [91]

The choice of the kernel function depends on the specific problem, the characteristics of the data, and the prior knowledge about the data distribution. The benefits of these kernels that they reduce the computational resources and time.

4.3.7.2.3 SVM process

- Import the dataset
- Explore the data to figure out what they look like
- Pre-process the data
- Split the data into attributes and labels
- Divide the data into training and testing sets
- Train the SVM algorithm
- Make some predictions
- Evaluate the results of the algorithm

4.3.7.2.4 Advantages and disadvantages

Table IV.3. SVM advantages and disadvantages

Advantages	Disadvantages
Memory Efficiency	Choosing a good kernel function is not easy
Effective in high dimensional spaces	Sensitivity to Noise
Works well with even unstructured and semi structured data	Parameter Tuning (C parameter for example)
Versatile Kernels	Long training time for large datasets
Robust to overfitting	In cases where the number of features for each data point exceeds the number of training data samples, the SVM will underperform

Overall, SVMs are a powerful and versatile algorithm that can be used for a variety of tasks such as recommendation systems, anomaly detection, image recognition, **healthcare** and text classification. However, they can be computationally expensive to train, and they may not be the best choice for all problems. It's important to note that the suitability of SVMs depends on the specific problem at hand and the characteristics of the dataset. It's recommended to experiment with different algorithms and evaluate their performance before selecting SVMs as the final choice.

4.4 Conclusion

The classification chapter provides a foundation for understanding and applying machine learning techniques to categorize data based on observed features. We explored algorithms, dimension reduction methods, evaluation metrics, and challenges in classification.

In the next chapter, we will present and analyze the results of our classification experiments. We will discuss performance metrics, compare algorithms, and explore the implications of dimension reduction. This analysis will help draw meaningful conclusions, identify areas for improvement, and propose future research directions.

The results and discussion chapter will deepen our understanding of classification performance and provide insights for real-world applications. It builds upon the knowledge gained in this chapter to effectively interpret the outcomes of our experiments.

V. Chapter 5 Results and discussion

5.1 Introduction

The findings and analysis of the experiments conducted in the study needs to be evaluated. This chapter aims to provide a comprehensive overview of the obtained results, evaluate their significance, and discuss their implications. The results are presented in a clear and organized manner, accompanied by relevant statistical measures and visual representations such as tables, figures (histograms).

The chapter provides a brief overview of the methodology employed, including the dataset used, the classifiers utilized, and the evaluation metrics employed to assess the performance of the models. Next, it delves into the detailed presentation of the results starting by presenting the results obtained from the experiments conducted on the original images, providing insights into the performance of different classifiers and the impact of various techniques such as dimension reduction and feature selection. The results are analyzed and compared, highlighting any notable trends or patterns observed. Then, the chapter moves on to discuss the results obtained from the experiments conducted on images and ROI images with feature extraction techniques applied, LBP in our case. It examines the performance of different classifiers, the influence of dimension reduction and feature selection techniques, and any significant findings or improvements achieved.

Throughout the presentation of results, appropriate statistical measures are provided to quantify the performance of the models, such as accuracy, precision, recall, f1-score, sensitivity, specificity, and average time. Visual representations, such as tables and figures, are used to enhance the understanding and clarity of the findings.

Following the presentation of results, a comprehensive discussion ensues. This section interprets the findings of the research objectives. It provides explanations for observed trends or patterns and discusses the practical implications of the results.

Chapter plays a critical role in the research process as it enables the researcher to draw meaningful conclusions, validate or reject the research hypotheses, and contribute to the existing body of knowledge in the field.

5.2 Tools and materials

5.2.1 Python

Python is a programming language that offers an interpreted, object-oriented approach with dynamic semantics. It boasts high-level built-in data structures and features dynamic typing and binding, making it a compelling choice for Rapid Application Development. It can also serve as a scripting or glue language, facilitating the connection of various existing components. Python's syntax is simple and easy to learn, prioritizing readability and reducing program maintenance costs. It supports modules and packages, promoting code modularity and reuse. The Python interpreter and its comprehensive standard library are freely available in both source and binary formats for major platforms, allowing for widespread distribution [92]. Python is used for web development, data analysis, artificial intelligence, machine learning, automation, and much more.

5.2.2 Anaconda

Anaconda is a data science-oriented distribution of the Python and R programming languages that provides an open-source solution for simplified package management and deployment. The management of package versions in Anaconda is handled by conda, a package management system that ensures the analysis of the current environment before performing installations, thereby avoiding disruptions to existing frameworks and packages.

With the Anaconda distribution, users gain access to more than 250 pre-installed packages. Additionally, they have the option to install over 7500 additional open-source packages from PyPI (Python Package Index) and the conda package and virtual environment manager. To enhance user experience, Anaconda includes a graphical user interface (GUI) called Anaconda Navigator, which serves as an alternative to the command line interface. Navigator, integrated within the Anaconda distribution, allows users to launch applications, manage conda packages, environments, and channels without relying on command-line commands. Through Navigator, users can search for packages, install them within a specific environment, execute the packages, and keep them up to date [93]. When installing Anaconda, Jupyter Notebook is automatically installed along with it. This means that once you have Anaconda installed on your system, you can start using Jupyter Notebook without any additional steps.

5.2.3 Jupyter notebook

Jupyter Notebook is a web-based interactive application designed for creating and sharing computational documents. Originally named IPython, the project underwent a rebranding in 2014 and became Jupyter. It is a fully open-source tool that offers all its functionalities for free. Jupyter Notebook supports a wide range of languages, exceeding 40 options, including popular choices like Python, R, and Scala. The core component of Jupyter Notebook is the notebook itself, which is a flexible file format saved in the ipynb format. Users can create, edit, and save notebooks containing code, text explanations, equations, and visualizations. To facilitate notebook management, Jupyter provides a notebook dashboard where users can organize and access multiple notebooks efficiently. One key feature of Jupyter Notebook is its support for kernels. Kernels are processes that execute interactive code in a specific programming language and deliver the resulting output to the user. They also respond to tab completion and introspection requests, enhancing the interactive coding experience. Jupyter notebooks offer the ability to convert them to various open standard formats, including HTML, LaTeX, PDF, Markdown, and Python files. This conversion process can be performed using the "Download As" function in the web interface. Additionally, automated conversion can be achieved using tools such as nbconvert, allowing for batch conversions or integration into automated workflows.

Jupyter notebooks serve multiple purposes and find applications in various domains. A notebook provides an interactive computational environment where users can execute specific code segments, observe the output, and modify the code iteratively to achieve desired results or explore further. Data exploration tasks benefit greatly from Jupyter notebooks due to their iterative nature. Additionally, Jupyter notebooks are extensively utilized in data science workflows, including machine learning experiments and modeling. They also serve as a platform for documenting code samples. Within a Jupyter notebook, users can run independent code cells in any order they prefer. Furthermore, documentation can be accomplished by seamlessly alternating between code cells and markdown cells for clear explanations and instructions [94].

5.2.4 Libraries

5.2.4.1 PIL Library

The Python Imaging Library (PIL) adds image processing capabilities to your Python interpreter. This library provides extensive file format support, an efficient internal representation, and fairly powerful image processing capabilities. The core image library is designed for fast access to data stored in a few basic pixel formats. It should provide a solid foundation for a general image processing tool [95]. there is a fork called "Pillow" that is a drop-in replacement for PIL and provides continued development and support. Pillow offers the same functionality as PIL and is compatible with the latest versions of Python.

5.2.4.2 OpenCV

OpenCV (Open-Source Computer Vision Library) is an open-source library that includes several hundreds of computer vision algorithms. The cv2 (OpenCV) provides a comprehensive set of functions and algorithms that enable developers to work with images, videos, and real-time computer vision applications. It has a modular structure, which means that the package includes several shared or static libraries [96]. The following modules are available:

- ***Core functionality (core):*** a compact module defining basic data structures, including the dense multi-dimensional array Mat and basic functions used by all other modules.
- ***Image Processing (imgproc):*** An image processing module that includes linear and non-linear image filtering, geometrical image transformations (resize, affine and perspective warping, generic table-based remapping), color space conversion, histograms, and so on.
- **Video Analysis (video):** a video analysis module that includes motion estimation, background subtraction, and object tracking algorithms.
- **Camera Calibration and 3D Reconstruction (calib3d):** Basic multiple-view geometry algorithms, single and stereo camera calibration, object pose estimation, stereo correspondence algorithms, and elements of 3D reconstruction.
- **2D Features Framework (features2d):** Salient feature detectors, descriptors, and descriptor matchers.
- **Object Detection (objdetect):** Detection of objects and instances of the predefined classes (for example, faces, eyes, mugs, people, cars, and so on).
- **High-level GUI (highgui):** An easy-to-use interface to simple UI capabilities.

- **Video I/O (videoio):** An easy-to-use interface to video capturing and video codecs.
- ... some other helper modules, such as FLANN and Google test wrappers, Python bindings, and others.

5.2.4.3 *OS*

OS, which stands for "Operating System" in Python provides functions for interacting with the operating system. It comes under Python's standard utility modules. This module provides a portable way of using operating system-dependent functionality to interact with the file system [97]. These are some functionalities of the OS library:

- **File and Directory Operations:** The "os" library provides functions for working with files and directories. You can create, delete, rename, and traverse directories using functions like os.mkdir(), os.rmdir(), os.rename(), os.listdir(), and more. It also offers file-related operations, including opening, closing, deleting, and checking file existence.
- ***Path Manipulation:*** The "os" library offers methods for manipulating file paths, making it easier to work with different operating systems and handle path-related operations. You can join paths using **os.path.join()**, split paths into directory and filename using os.path.split(), get the basename or directory name using os.path.basename() and os.path.dirname(), and more.
- **Process Management:** The "os" library allows you to manage processes from within your Python script. You can launch external programs, run system commands, and handle process-related operations using functions like os.system(), os.popen(), and os.spawn(). It also provides functionality for working with process IDs, process termination, and process-related information.
- **Environment Variables:** The "os" library enables you to access and modify environment variables of the operating system. You can retrieve individual environment variables using os.getenv(), get a dictionary of all environment variables using os.environ, and modify or create new environment variables using os.putenv().
- **Miscellaneous:** The "os" library includes additional functionalities, such as working with file permissions, getting the current working directory, changing the current working directory using os.chdir(), getting system information using os.uname() (on Unix-based systems), and more.

5.2.4.4 NumPy

NumPy is a Python library that stands for "Numerical Python." It is a fundamental package for scientific computing and data manipulation in Python. NumPy provides a powerful N-dimensional array object, which allows efficient storage and manipulation of large, homogeneous datasets. It includes a vast collection of mathematical functions to operate on arrays, enabling numerical operations, linear algebra calculations, random number generation, and much more. With its efficient array operations and broad range of functionality, NumPy serves as a fundamental building block for numerous scientific and data analysis libraries in Python. NumPy is a Python library that stands for "Numerical Python." It is a fundamental package for scientific computing and data manipulation in Python. NumPy provides a powerful N-dimensional array object, which allows efficient storage and manipulation of large, homogeneous datasets. It includes a vast collection of mathematical functions to operate on arrays, enabling numerical operations, linear algebra calculations, random number generation, and much more. With its efficient array operations and broad range of functionality, NumPy serves as a fundamental building block for numerous scientific and data analysis libraries in Python.

5.2.4.5 Matplotlib

Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python. Matplotlib makes easy things easy and hard things possible [98]. It provides a flexible and comprehensive set of tools for generating plots, charts, histograms, scatterplots, and other types of visualizations. These are some of its functionalities:

- It provides a diverse set of plotting functions that help you to create various types of visualizations, including line plots, bar plots, scatter plots, histograms, and pie charts. These functions empower you to visualize data, uncover patterns, trends, and relationships.
- You have extensive customization options in Matplotlib, allowing you to tailor your plots to meet specific requirements. You can customize elements such as colors, line styles, markers, labels, axes, titles, grids, and legends. It supports the use of different style sheets and themes, enabling quick changes to the appearance of your plots.
- It offers multiple interfaces for creating plots. The pyplot module, resembling MATLAB's interface, is commonly used for interactive plot creation. Additionally,

Matplotlib can be used with object-oriented APIs, where plots are generated using explicit figure and axis objects. This approach provides finer control over plot elements.

- Matplotlib seamlessly integrates with Jupyter Notebooks, making the creation and display of plots directly within notebook cells. This integration allows for interactive exploration and data analysis using visualizations.
- Matplotlib supports a wide range of output formats, including PNG, PDF, SVG, and more, enabling you to save plots in different file types. You can also copy plots to the clipboard or embed them in other applications. Matplotlib provides options for adjusting the resolution, size, and other properties of saved plots.
- It includes support for creating 3D visualizations such as 3D surfaces, scatter plots, and wireframes. These capabilities are valuable for visualizing data in three dimensions, making it useful in scientific computing, engineering, and data science.
- Being an open-source library, Matplotlib benefits from an active and supportive community. It offers a wide range of extensions and plugins developed by the community to enhance its functionalities. These additions provide additional plotting capabilities, styles, and tools.

Matplotlib is commonly used in data analysis, scientific research, engineering, machine learning, and other fields where data visualization is essential.

5.2.4.6 Scikit-learn

The scikit-learn library, often referred to as sklearn, is a popular machine learning library in Python. It is an open-source machine learning library that supports supervised and unsupervised learning. It also provides various tools for model fitting, data preprocessing, model selection, model evaluation, and many other utilities [99]. We can mention some tools used by scikit-library:

- ***Integration with NumPy and pandas:*** scikit-learn offers seamless integration with widely-used libraries like NumPy and pandas. It accepts NumPy arrays and pandas DataFrames as input, enabling effortless integration with data manipulation and preprocessing workflows. This integration facilitates the combination of scikit-learn with other Python libraries for data analysis and processing, providing a convenient and powerful environment for machine learning tasks.
- ***Data preprocessing and Feature extraction:*** The sklearn.preprocessing package offers a range of utility functions and transformer classes that are useful for transforming raw

feature vectors into a more suitable representation for downstream estimators. Standardizing the data set is generally beneficial for learning algorithms. In cases where outliers exist within the dataset, it is more appropriate to use robust scalers or transformers. The techniques used for preprocessing and feature extraction are: Standardization or mean removal and variance scaling, non-linear transformation, normalization, encoding categorical features, discretization, imputation of missing values, generating polynomial features, custom transformers. These techniques are crucial for preparing data before feeding it into machine learning models.

- ***Dimension reduction and Feature selection:*** Scikit-learn provides various techniques for dimensionality reduction, which refers to the process of reducing the number of features or variables in a dataset while preserving important information. Some of the dimensionality reduction techniques available in scikit-learn are: **Principal Component Analysis (PCA)**, Independent component analysis (ICA), **Random Projection (RP) including Sparse Random Projection (SRP)** and **Gaussian Random Projection (GRP),** Non-negative matrix factorization (NMF), **Linear Discriminant Analysis (LDA)**. These techniques are used as functions implemented in the Sklearn library to facilitates their use.

  For the feature selection task, Scikit-learn offers various feature selection techniques to help identify the most relevant and informative features from a dataset. These techniques aid in reducing the dimensionality of the data and improving model performance by focusing on the most significant features. From these techniques we can cite: Univariate Feature Selection, Recursive Feature Elimination (RFE), **SelectKBest**, L1 Regularization (Lasso) and Sequential Feature Selection (SFS). These are just a few examples of the feature selection techniques available in scikit-learn. Each technique has its own strengths and applicability depending on the data and the problem to solve.
- ***Classification and machine learning algorithms:*** A comprehensive set of supervised and unsupervised machine learning algorithms are included. Sklearn covers a wide range of algorithms such as: linear regression, logistic regression, decision trees, random forests, **support vector machines (SVM)**, **k-nearest neighbors (KNN)**, clustering algorithms and much more. These algorithms are designed with a consistent and easy-to-use interface, making the effort of switching between different algorithms easier for the experimentation and comparison purposes.
- ***Model Selection and Evaluation:*** Scikit-learn offers tools for model selection and evaluation, including techniques for cross-validation, hyperparameter tuning, and model

evaluation metrics. It provides methods for **splitting data into training and testing sets**, performing cross-validation to assess model performance, and optimizing model parameters to improve accuracy and generalization. The library includes a wide range of evaluation metrics such as **accuracy**, **precision**, **recall**, **F1-score** using **accuracy_score (), precision_score (), recall_score ()** and **f1_score () functions** respectively, ROC curves, and more. For evaluating the performance of the classifier used and for computing these metrics, Sklearn offers a function named **confusion matrix.**

Confusion matrix: The confusion_matrix () is a function which computes the confusion matrix with each row corresponding to the true class. For binary problems, we can get counts of **True Negatives**, **False Positives**, **False Negatives** and **True Positives**.

False Negative, False Positive, True Negative, and True Positive are terms used to describe the accuracy of a binary classification model, which classifies items into one of two categories.

- **False Negative (FN):** A false negative is an incorrect prediction that an item belongs to the negative class. In other words, the model predicts that an item belongs to the negative class when it actually belongs to the positive class.
- **False Positive (FP):** A false positive is an incorrect prediction that an item belongs to the positive class. In other words, the model predicts that an item belongs to the positive class when it actually belongs to the negative class.
- **True Negative (TN):** A true negative is a correct prediction that an item belongs to the negative class. In other words, the model correctly identifies an item as not belonging to the class it actually does not belong to.
- **True Positive (TP):** A true positive is a correct prediction that an item belongs to the positive class. In other words, the model correctly identifies an item as belonging to the class it actually belongs to.

These four metrics can be used to evaluate the accuracy of a binary classification model by calculating measures such as accuracy, precision, recall, and F1 score.

- **Community and Documentation:** scikit-learn has a strong community of developers and users who contribute to its development and provide support. The library is welldocumented, with a comprehensive user guide, API reference, and numerous examples and tutorials.

Sklearn is widely used for various machine learning tasks, including classification, regression, clustering, and dimensionality reduction. Its intuitive interface, wide range of supported algorithms and documentation which also provides some good examples make it as a preferred library for machine learning projects in Python.

5.2.4.7 *Time*

The time module in Python provides functions for handling time-related tasks. Some of the related time tasks include: reading the current time, formatting time, sleeping for a specified number of seconds and so on [100].

Some commonly used functions from the time module are: ctime (), sleep (), gmtime (), localtime () and **time ()** which returns the current time in seconds since the epoch (January 1, 1970, 00:00:00 UTC). It is often used for measuring elapsed time or benchmarking code. In this study, this function is used for calculating the average execution time.

5.2.5 MIAS database

In this study we used the mini-MIAS database which contains 322 digitized mammogram images consisted of left and right breast images. The acquired mammogram images are classified into three major cases: malignant, benign and normal. The size of these images is 1024 × 1024 pixels in Portable Greymap (PGM) format. Each pixel in the images is represented as a 8-bit word, where the images are in grayscale format with a pixel intensity of range [0, 255] [54]. Figure V.1 shows of different components in the image mammography.

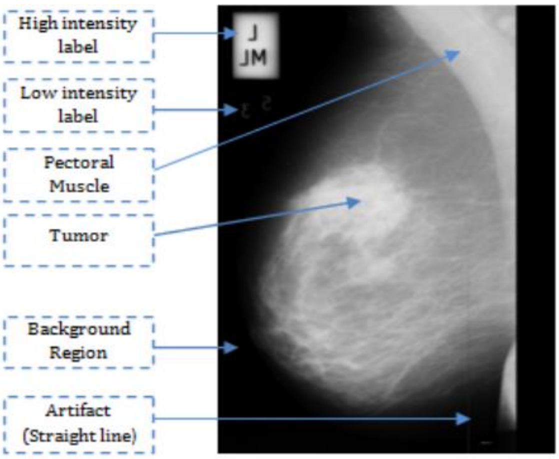


Figure V.12. Elements of mammogram images example [54]

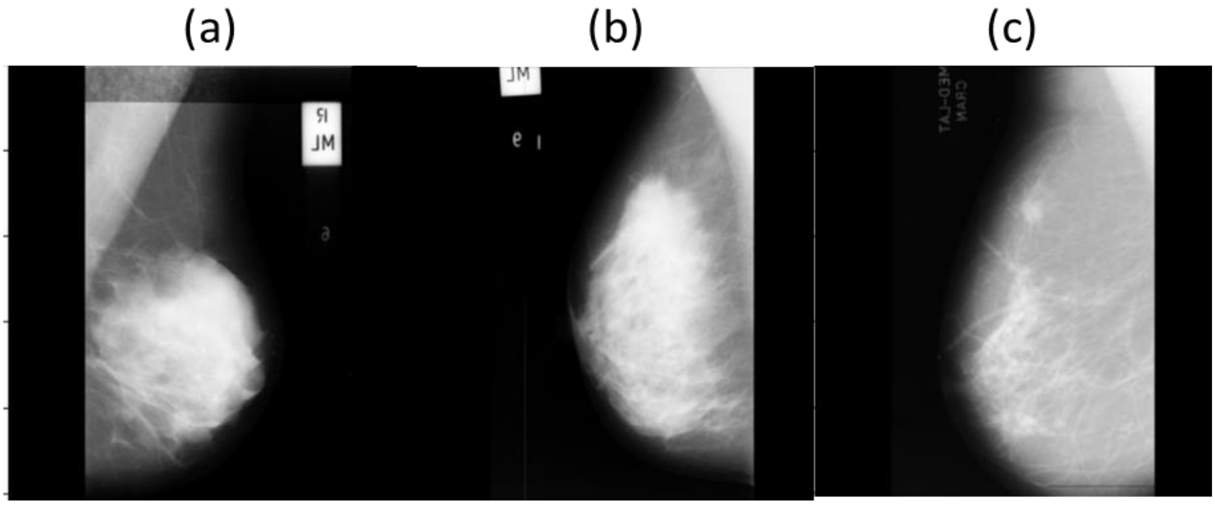


Figure V.13. (a) Benign, (b) Normal and (c) Malignant images from MIAS

5.2.5.1 Detailed information about MIAS

The follow list gives the films in the MIAS database and provides appropriate details as follows:

- **1st column:**

  MIAS database reference number.

- **2nd column:**

  Character of background tissue:

  F: Fatty

  G: Fatty-glandular

  D: Dense-glandular

- **3rd column:**

  Class of abnormality present:

  CALC: Calcification

  CIRC: Well-defined/circumscribed masses

  SPIC: Spiculated masses

  MISC: Other, ill-defined masses

  ARCH: Architectural distortion

  ASYM: Asymmetry

  NORM: Normal

- **4th column:**

  Severity of abnormality;

B: Benign

M: Malignant

- **5th, 6th columns:**

  x, y: image-coordinates of center of abnormality.

- **7th column:**

  Approximate radius (in pixels) of a circle enclosing the abnormality.[1]

5.3 Proposed method

In our study, we utilized the MIAS dataset as the input images for our classification task. We employed two classifiers, namely KNN and SVM, to perform the classification. For feature extraction, we adopted the Local Binary Patterns (LBP) technique and focused on extracting the region of interest (ROI) from the images. SelectKBest was employed for feature selection to identify the most informative features. To compare different dimension reduction methods, we employed PCA, GRP, SRP, and LDA. Our study consisted of three main steps. In the first step, we divided it into four sub-steps: (1) classifying the original images, (2) applying dimension reduction techniques, (3) applying feature selection techniques, and (4) employing both feature selection and dimension reduction using both SVM and KNN. The second step mirrored the first one, but this time we applied LBP on the images. Finally, in the third step, we repeated the process using LBP on the extracted ROI of the images. Through these steps, we aimed to evaluate the impact of various techniques on the classification performance. In summary, our study conducted a comprehensive analysis using the MIAS dataset, involving various classifiers, feature extraction techniques, feature selection, and dimension reduction methods. The aim of this study is to develop a predictive model that can accurately classify images as either normal or abnormal (binary classification). The model is trained on a dataset of images and learns patterns and features that distinguish between normal and abnormal images. By leveraging various techniques such as feature extraction, dimension reduction, and classification algorithms, the study aims to create a robust and accurate model that can effectively predict the classification of new images. The ultimate goal is to provide a valuable tool for automated image analysis and assist in diagnosing abnormalities in medical imaging. The parameters used and the applied metrics for evaluating the performance are discussed next.

[1] For more details you can visit the official MIAS website via this link: http://peipa.essex.ac.uk/info/mias.html

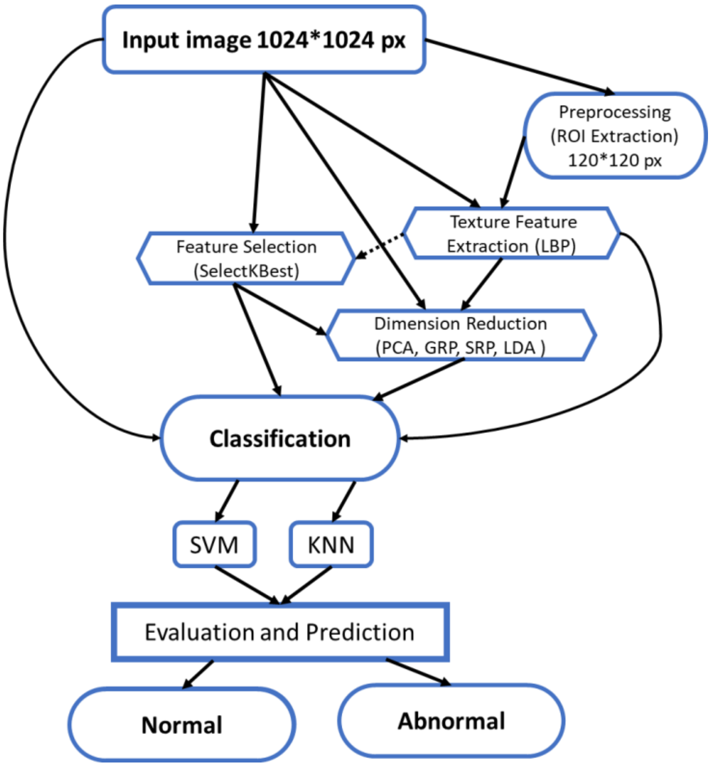

Input image 1024*1024 px
Preprocessing (ROI Extraction) 120*120 px
Texture Feature Extraction (LBP)
Feature Selection (SelectKBest)
Dimension Reduction (PCA, GRP, SRP, LDA)
Classification
SVM
KNN
Evaluation and Prediction
Normal
Abnormal

Figure V.14. Applicated methods

5.4 Results

To evaluate the performance of our proposed method, we employed a comprehensive set of evaluation metrics including accuracy, precision, recall, F1-score, sensitivity, and specificity. These metrics provide a comprehensive assessment of the classification performance, taking into account different aspects such as correct predictions, false positives, false negatives, true negatives and true positives. In addition to the evaluation metrics, we also considered the execution time to analyze and compare the computational efficiency of the methods used. By considering the execution time, we gained insights into the practical feasibility of the proposed approach, especially in scenarios where real-time or time-sensitive processing is required. The goal of using these evaluations metrics is to take into account both the quality of the results and the practical considerations of computational efficiency.

5.4.1 Parameters

After conducting an extensive experimentation process, we arrived at the following optimal parameter values which are constant for our proposed method:

Table V.4. Fixed parameters for our experiments

Parameters	Values
K (KNN)	9
Random state (KNN and SVM)	27
Random state (PCA, RP, LDA)	42
Train and test (KNN and SVM)	80% / 20%
C (SVM)	2.1
Kernels (SVM)	Sigmoid, Polynomial and radial basis function (RBF)

For the feature selection technique used (SelectKBest) with f_classif function, here are the values of K which represents the selected features for our experiments:

Table V.5. Values of K (SelectKBest) used in our investigation

Parameters	Values
K (original images)	617487
K (Images with LBP)	783
K (ROI of images with LBP)	118

The accuracy, precision, recall, f1-score, sensitivity and specificity are given as follow:

- **Accuracy**: The accuracy measures the overall correctness of the classification predictions.

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} * 100 \tag{35}$$

- **Precision:** The precision measures the proportion of correctly predicted positive instances out of the total predicted positive instances.

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} * 100 \tag{36}$$

- **Recall:** The recall measures the proportion of correctly predicted positive instances out of the total actual positive instances.

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} * 100 \tag{37}$$

- **F1-score:** The F1-score is the harmonic mean of precision and recall. It provides a balanced measure of precision and recall.

$$\text{F1-score} = 2 * \frac{\text{Precision}*\text{Recall}}{\text{Precision+Recall}} * 100 \tag{38}$$

- **Sensitivity:** Sensitivity, also known as True Positive Rate (TPR), measures the proportion of actual positive instances that are correctly predicted as positive by the classifier.

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP+FN}} * 100 \tag{39}$$

- **Specificity:** Specificity, also known as True Negative Rate (TNR), measures the proportion of actual negative instances that are correctly predicted as negative by the classifier.

$$\text{Specificity} = \frac{\text{TN}}{\text{TN+FP}} * 100 \tag{40}$$

Where TN, TP, FN, FP are: true negatives, true positives, false negatives and false positives respectively.

5.4.2 Tables of results

The results of our classification using the proposed method are shown in the tables below:

Table V.6. Classification results for the original MIAS images

Classifiers	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	Sensitivity (%)	Specificity (%)	Average time (s)
1. Original images							
1.1 KNN							
KNN	75.38	75.30	75.38	71.56	95.56	30	2.45
KNN + PCA (nc=114)	76.92	78.89	76.92	72.82	97.78	30	36.42
KNN + SRP (nc=107)	**78.46**	**78.74**	**78.46**	**75.95**	**95.56**	**40**	**2.10**
KNN + GRP (nc=112)	76.92	77.09	76.92	73.80	95.56	35	5.43
KNN + LDA (nc=1)	64.62	64.13	64.62	64.36	75.56	40	75.68
1.2 KNN + SelectKBest							
KNN	75.38	77.27	75.38	70.41	97.78	25	2.90
KNN + PCA (nc=31)	75.38	74.41	75.38	72.51	93.33	35	10.16
KNN + SRP (nc=152)	**78.46**	**80.36**	**78.46**	**75.11**	**97.78**	**35**	**2.55**
KNN + GRP(nc=107)	**78.46**	**78.74**	**78.46**	**75.49**	**95.56**	**40**	**4.69**
KNN + LDA (nc=1)	64.62	63.15	64.62	63.75	77.78	35	45.10
1.3 SVM							
1.3.1 Sigmoid kernel							
SVM	75.38	75.30	75.38	71.56	95.56	30	45.43
SVM + PCA (nc=47)	70.77	79.45	70.77	60.09	100	5	19.23
SVM + SRP (nc=32)	**76.92**	**78.89**	**76.92**	**72.82**	**97.78**	**30**	**1.74**
SVM + GRP (nc=89)	70.77	79.45	70.77	60.09	100	5	4.51
SVM + LDA (nc=1)	63.08	60.99	63.08	61.81	77.78	30	75.49
1.3.2 Polynomial kernel							
SVM	63.08	58.59	63.08	59.97	82.22	20	47.23
SVM + PCA (nc=48)	72.31	70.24	72.31	68	93.33	25	19.03
SVM + SRP (nc=28)	72.31	80.22	72.31	63.29	100	10	1.74
SVM + GRP (nc=34)	72.31	73.01	72.31	65.17	97.78	15	2.57

SVM + LDA (nc=1)	64.62	46.90	64.62	54.35	93.33	0	75.54
1.3.3 Rbf kernel							
SVM	69.23	78.70	69.23	56.64	100	0	99.57
SVM + PCA (nc=1)	69.23	78.70	69.23	56.64	100	0	12.33
SVM + SRP (nc=1)	69.23	78.70	69.23	56.64	100	0	1.61
SVM + GRP (nc=36)	70.77	79.45	70.77	60.09	100	5	2.61
SVM + LDA (nc=1)	66.15	65.23	66.15	65.63	77.78	40	75.85
1.4 SVM + SelectKbest							
1.4.1 Sigmoid kernel							
SVM	66.15	65.23	66.15	65.63	77.78	40	23.78
SVM + PCA (nc=12)	70.77	68.08	70.77	64.09	95.56	15	11.05
SVM + SRP (nc=39)	75.38	81.84	75.38	69.04	100	20	2.36
SVM + GRP (nc=30)	73.85	72.14	73.85	71.24	91.11	35	3.06
SVM + LDA (nc=1)	64.62	61.09	64.62	62.14	82.22	25	46.85
1.4.2 Polynomial kernel							
SVM	66.15	63.41	66.15	64.23	82.22	30	27.99
SVM + PCA (nc=34)	73.85	73.30	73.85	69.20	95.96	25	10.89
SVM + SRP (nc=4)	72.31	80.22	72.31	63.29	100	10	2.28
SVM + GRP (nc=2)	69.23	78.70	69.23	56.64	100	0	2.32
SVM + LDA (nc=1)	63.08	52.06	63.08	55.62	88.89	5	48.51
1.4.3 Rbf kernel							
SVM	66.15	47.25	66.15	55.13	95.56	0	69.48
SVM + PCA (nc=7)	70.77	79.45	70.77	60.09	100	5	9.26
SVM + SRP (nc=1)	69.23	78.70	69.23	56.64	100	0	2.25
SVM + GRP (nc=2)	70.77	79.45	70.77	60.09	100	5	2.35
SVM + LDA (nc=1)	66.15	65.23	66.15	65.63	77.78	40	48.09

Table V.7. Classification results for MIAS images after applying LBP

Classifiers	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1-score (%)**	**Sensitivity (%)**	**Specificity (%)**	**Average time (s)**
1. Images with LBP							
1.1 KNN							
KNN	61.54	46.15	61.54	52.75	88.89	0	2.52
KNN + PCA (nc=49)	73.85	73.30	73.85	69.20	95.56	25	20.24
KNN + SRP (nc=96)	75.38	75.38	75.38	75.38	82.22	60	2.18
KNN + GRP (nc=75)	78.46	83.57	78.46	74.11	100	30	4.34
KNN + LDA (nc=1)	67.69	47.60	67.69	55.89	97.78	0	77.40
1.2 KNN + SelectKBest							
KNN	78.46	83.57	78.46	74.11	100	30	0.01
KNN + PCA (nc=3)	**96.92**	**97.05**	**96.92**	**96.88**	**100**	**90**	**0.02**
KNN + SRP (nc=154)	90.77	91.86	90.77	90.24	100	70	0.02
KNN + GRP (nc=49)	84.62	85.68	84.62	83.32	97.78	55	0.02
KNN + LDA (nc=1)	95.38	95.37	95.38	95.35	97.78	90	0.1
1.3 SVM							
1.3.1 Sigmoid kernel							
SVM	69.23	78.70	69.23	56.64	100	0	44.97
SVM + PCA (nc=103)	70.77	79.45	70.77	60.09	100	5	33.89
SVM + SRP (nc=7)	69.23	65.18	69.23	64.45	91.11	20	1.74
SVM + GRP (nc=2)	70.77	68.08	70.77	64.09	95.56	15	1.42
SVM + LDA (nc=1)	60	52.73	60	55.33	82.22	10	73.26
1.3.2 Polynomial kernel							
SVM	69.23	78.70	69.23	56.64	100	0	59.34
SVM + PCA (nc=31)	70.77	79.45	70.77	60.09	100	5	15.75
SVM + SRP (nc=127)	72.31	80.22	72.31	63.29	100	10	2.03
SVM + GRP (nc=3)	70.77	79.45	70.77	60.09	100	5	1.61
SVM + LDA (nc=1)	69.23	78.70	69.23	56.64	100	0	77.39
1.3.3 Rbf kernel							
SVM	69.23	78.70	69.23	56.64	100	0	126.37

SVM + PCA (nc=137)	72.31	80.22	72.31	63.29	100	10	44.42
SVM + SRP (nc=1)	69.23	78.70	69.23	56.64	100	0	1.64
SVM + GRP (nc=2)	69.23	78.70	69.23	56.64	100	0	1.46
SVM + LDA (nc=1)	69.23	78.70	69.23	56.64	100	0	77.28
1.4 SVM + SelectKbest							
1.4.1 Sigmoid kernel							
SVM	69.23	78.70	69.23	56.64	100	0	0.03
SVM + PCA (nc=57)	**100**	**100**	**100**	**100**	**100**	**100**	**0.03**
SVM + SRP (nc=167)	93.85	94.35	93.85	93.64	100	80	0.03
SVM + GRP (nc=196)	92.31	93.08	92.31	91.96	100	75	0.03
SVM + LDA (nc=1)	93.85	94.19	93.85	93.92	93.33	95	0.09
1.4.2 Polynomial kernel							
SVM	**100**	**100**	**100**	**100**	**100**	**100**	**0.04**
SVM + PCA (nc=7)	98.46	98.49	98.46	98.45	100	95	0.03
SVM + SRP (nc=77)	**100**	**100**	**100**	**100**	**100**	**100**	**0.02**
SVM + GRP(nc=186)	**100**	**100**	**100**	**100**	**100**	**100**	**0.03**
SVM + LDA (nc=1)	95.38	95.67	95.38	95.27	100	85	0.09
1.4.3 Rbf kernel							
SVM	**100**	**100**	**100**	**100**	**100**	**100**	**0.04**
SVM + PCA (nc=3)	**100**	**100**	**100**	**100**	**100**	**100**	**0.02**
SVM + SRP (nc=155)	**100**	**100**	**100**	**100**	**100**	**100**	**0.02**
SVM + GRP (nc=158)	98,46	98,49	98,46	98,45	100	95	0.03
SVM + LDA (nc=1)	93.85	93.85	93.85	93.85	95.56	90	0.08

Table V.8. Classification results for Region of Interest of MIAS images (ROI) after applying LBP

Classifiers	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1-score (%)**	**Sensitivity (%)**	**Specificity (%)**	**Average time (s)**
1. ROI Images with LBP							
1.1 KNN							
KNN	75.38	74.49	75.38	74.62	86.36	52.38	0.04
KNN + PCA (nc=171)	81.54	81.12	81.54	80.97	90.91	61.90	0.68
KNN + SRP (nc=66)	80	79.80	80	78.74	93.18	52.38	0.06
KNN + GRP (nc=1)	72.31	72.10	72.31	67.29	95.45	23.81	0.03
KNN + LDA (nc=1)	70.77	68.90	70.77	68.92	86.36	38.10	1
1.2 KNN + SelectKBest							
KNN	87.69	87.69	87.69	87.69	90.91	80.95	0.02
KNN + PCA (nc=43)	**92.31**	**92.26**	**92.31**	**92.26**	**95.45**	**85.71**	**0.02**
KNN + SRP (nc=31)	87.69	87.54	87.69	87.52	93.18	76.19	0.01
KNN + GRP (nc=47)	90.77	91.05	90.77	90.48	97.73	76.19	0.01
KNN + LDA (nc=1)	80	79.78	80	79.87	86.36	66.67	0.03
1.3 SVM							
1.3.1 Sigmoid kernel							
SVM	67.69	78.13	67.69	54.65	100	0	0.59
SVM + PCA (nc=112)	81.54	82.01	81.54	80.14	95.45	52.38	0.55
SVM + SRP (nc=29)	70.77	71.95	70.77	63.20	97.73	14.29	0.06
SVM + GRP (nc=14)	69.23	78.85	69.23	58.09	100	4.76	0.07
SVM + LDA (nc=1)	53.85	56.03	53.85	54.75	61.36	38.10	0.81
1.3.2 Polynomial kernel							
SVM	75.38	74.89	75.38	72.79	93.18	38.10	0.75
SVM + PCA (nc=2)	69.23	78.85	69.23	58.09	100	4.76	0.22
SVM + SRP (nc=79)	76.92	76.17	76.92	75.47	90.91	47.62	0.09
SVM + GRP (nc=46)	73.85	74.11	73.85	69.67	95.45	28.57	0.08
SVM + LDA (nc=1)	69.23	78.85	69.23	58.09	100	4.76	0.77
1.3.3 Rbf kernel							

SVM	76.92	79.34	76.92	73.24	97.73	33.33	1.01
SVM + PCA (nc=131)	81.54	82.01	81.54	80.14	95.45	52.38	0.65
SVM + SRP (nc=79)	78.46	80.70	78.46	75.45	97.73	38.10	0.09
SVM + GRP (nc=21)	70.77	79.58	70.77	61.29	100	9.52	0.06
SVM + LDA (nc=1)	78.46	79.08	78.46	76.20	95.45	42.86	0.76
1.4 SVM + SelectKbest							
1.4.1 Sigmoid kernel							
SVM	67.69	78.13	67.69	54.65	100	0	0.02
SVM + PCA (nc=24)	87.69	87.74	87.69	87.31	95.45	71.43	0.03
SVM + SRP (nc=92)	86.15	87.07	86.15	85.28	97.73	61.90	0.03
SVM + GRP (nc=90)	84.62	85.80	84.6	83.45	97.73	57.14	0.02
SVM + LDA (nc=1)	72.31	73.94	72.31	72.85	75	66.67	0.08
1.4.2 Polynomial kernel							
SVM	75.38	76.08	75.38	75.66	79.55	66.67	0.02
SVM + PCA (nc=4)	83.08	84.54	83.08	81.55	97.73	52.38	0.02
SVM + SRP (nc=41)	89.23	89.69	89.23	88.79	97.73	71.43	0.01
SVM + GRP (nc=35)	89.23	89.69	89.23	88.79	97.73	71.43	0.01
SVM + LDA (nc=1)	78.46	77.81	78.46	77.80	88.64	57.14	0.02
1.4.3 Rbf kernel							
SVM	84.62	84.43	84.62	84.14	93.18	66.67	0.02
SVM + PCA (nc=72)	89.23	89.69	89.23	88.79	97.73	71.43	0.03
SVM + SRP (nc=41)	90.77	91.88	90.77	90.29	100	71.43	0.02
SVM + GRP (nc=41)	**92.31**	**92.44**	**92.31**	**92.14**	**97.73**	**80.95**	**0.02**
SVM + LDA (nc=1)	80	79.78	80	79.87	86.36	66.67	0.02

5.4.3 Histograms representing the results

The constructed histograms represent the accuracy and precision values achieved by various methods. The X-axis represents the different methods employed, while the Y-axis represents the corresponding accuracy and precision values. The histograms provide a visual comparison of the performance of each method in terms of accuracy and precision.

5.4.3.1 *Original Images*

- **KNN**

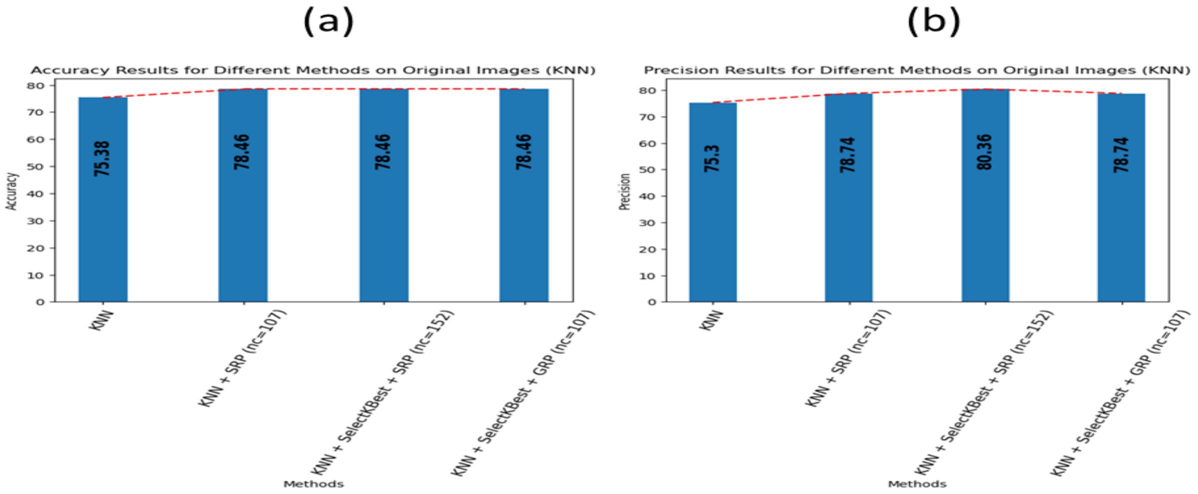


Figure V.15. (a) Accuracy, (b) Precision results on original images using different methods with KNN

- **SVM**

1. **Sigmoid kernel**

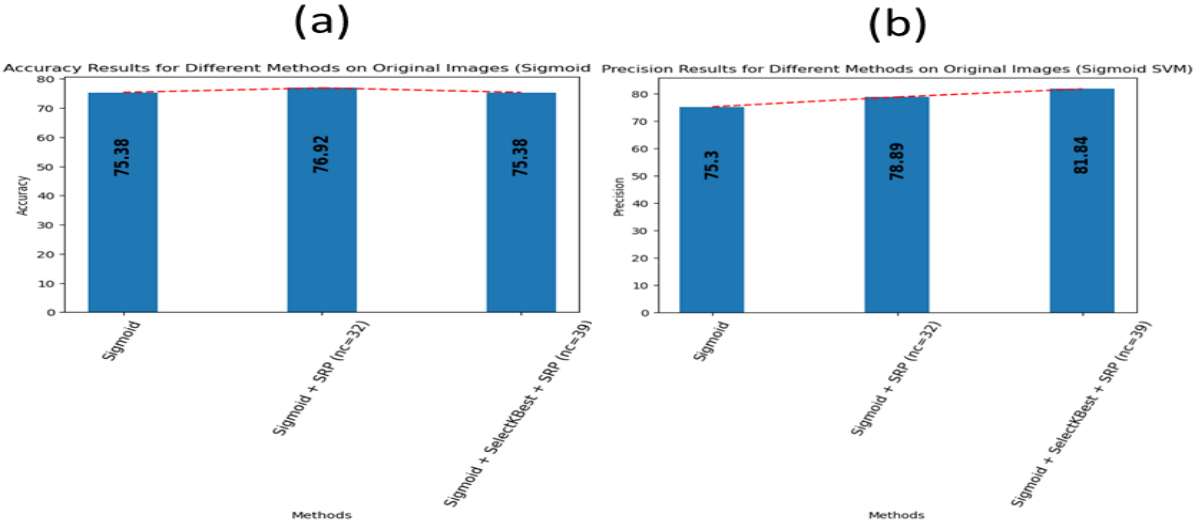


Figure V.16. (a) Accuracy, (b) Precision results on original images using different methods with Sigmoid SVM

2. Polynomial kernel

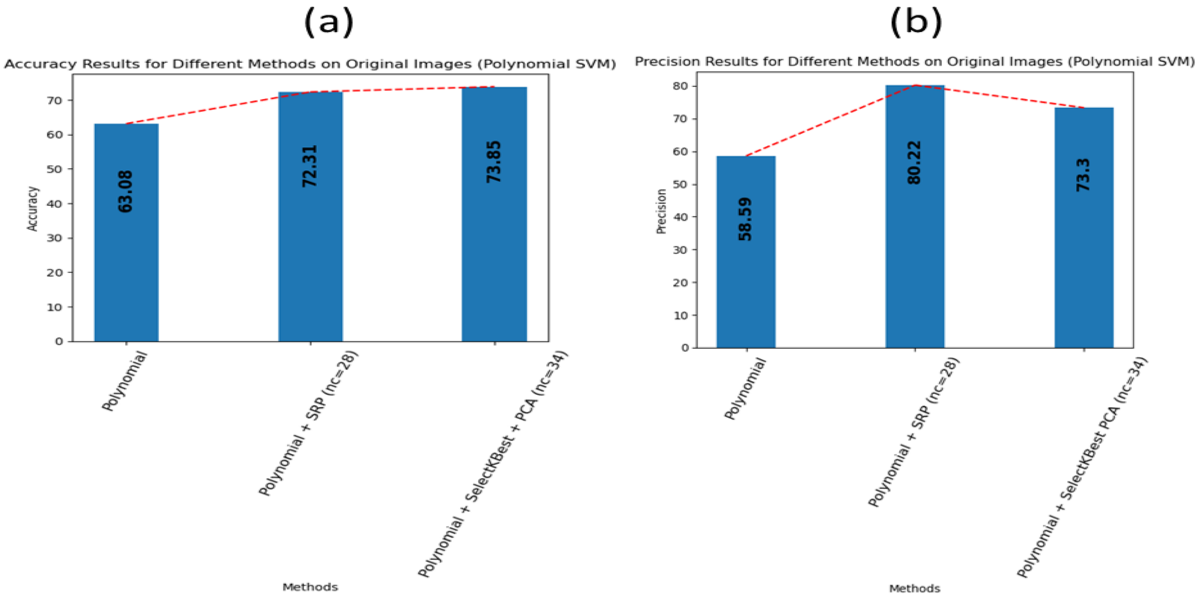


Figure V.17. Accuracy, (b) Precision results on original images using different methods with Polynomial SVM

3. RBF kernel

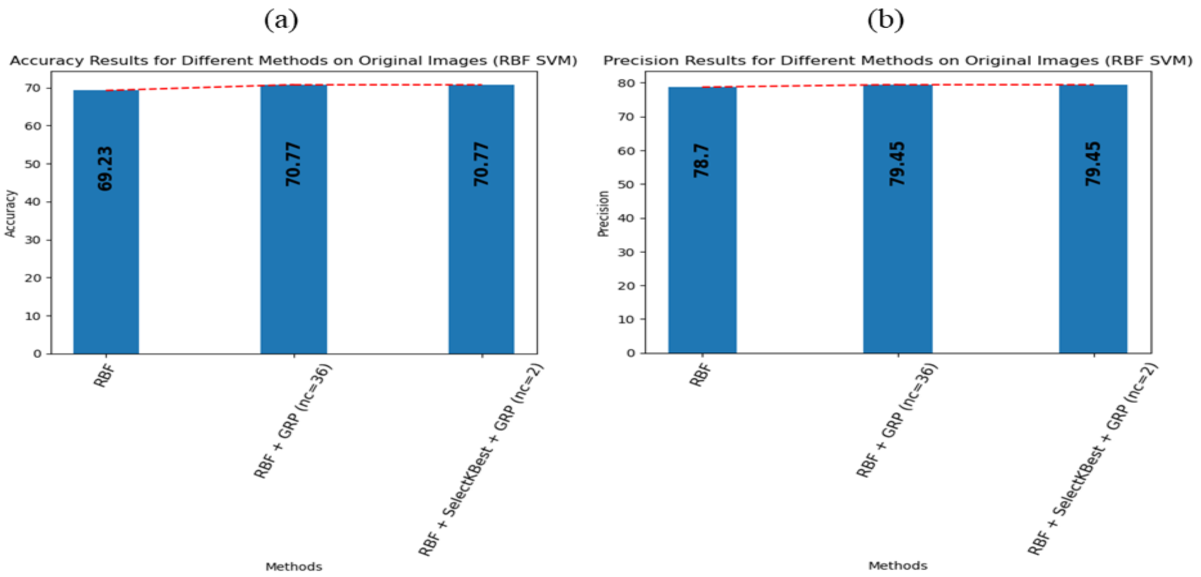


Figure V.18. Accuracy, (b) Precision results on original images using different methods with RBF SVM

5.4.3.2 Images with LBP (feature extraction)

- **KNN**

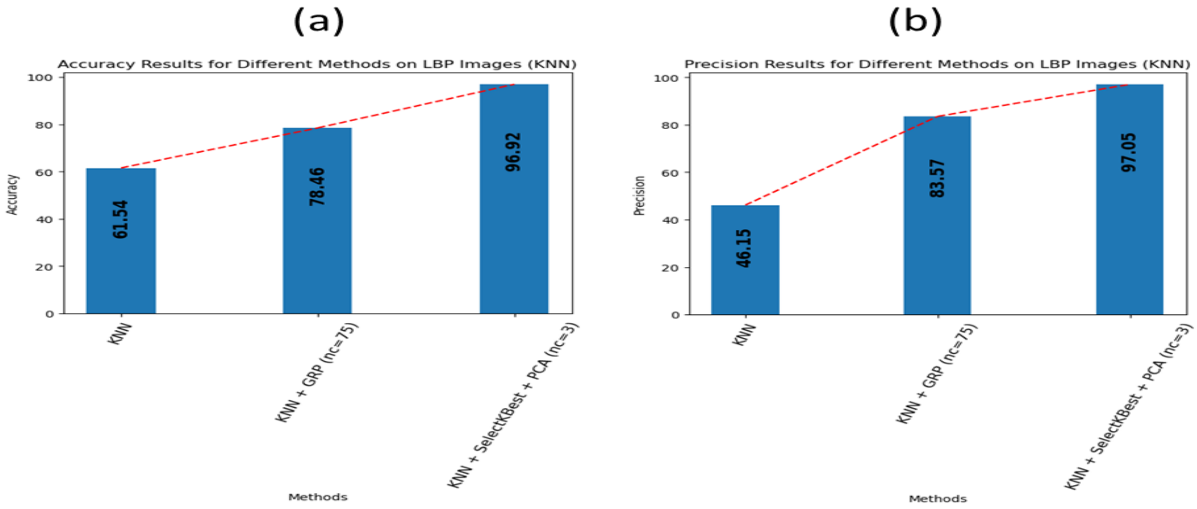


Figure V.19. (a) Accuracy, (b) Precision results on images with LBP using different methods with KNN

- **SVM**

  1. **Sigmoid kernel**

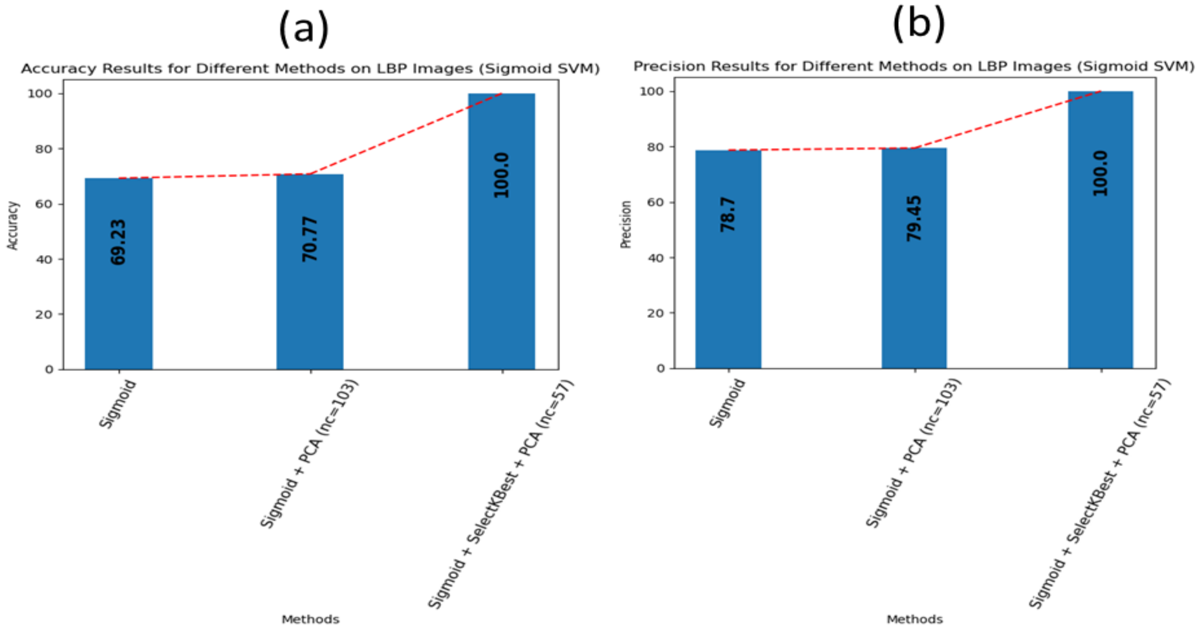


Figure V.20. (a) Accuracy, (b) Precision results on images with LBP using different methods with Sigmoid SVM

2. Polynomial kernel

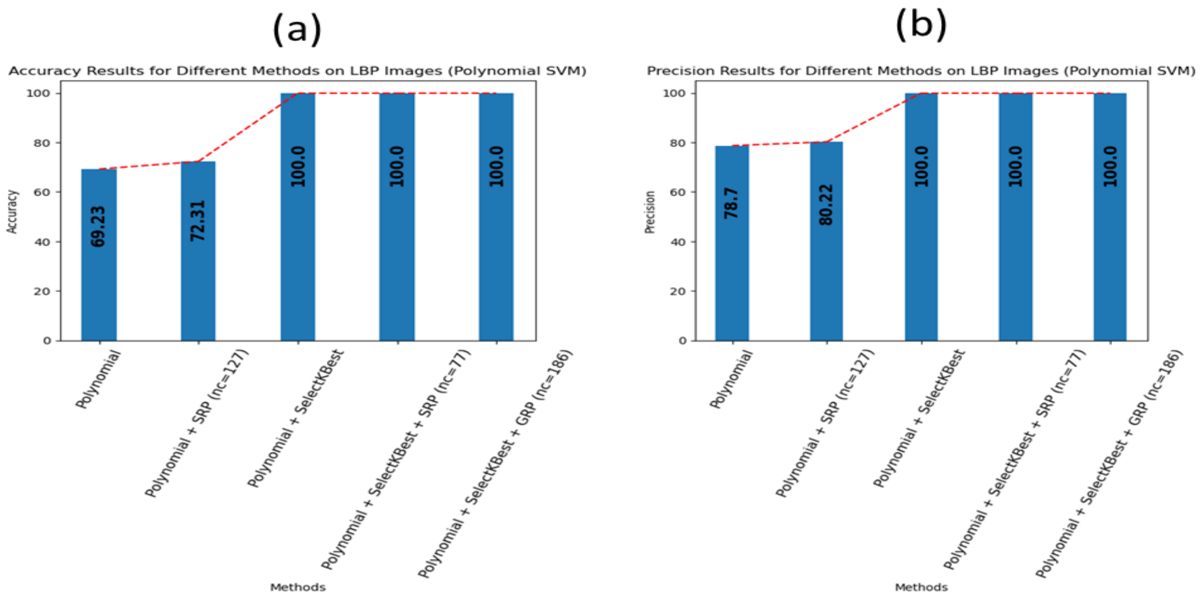


Figure V.21. (a) Accuracy, (b) Precision results on images with LBP using different methods with Polynomial SVM

3. RBF kernel

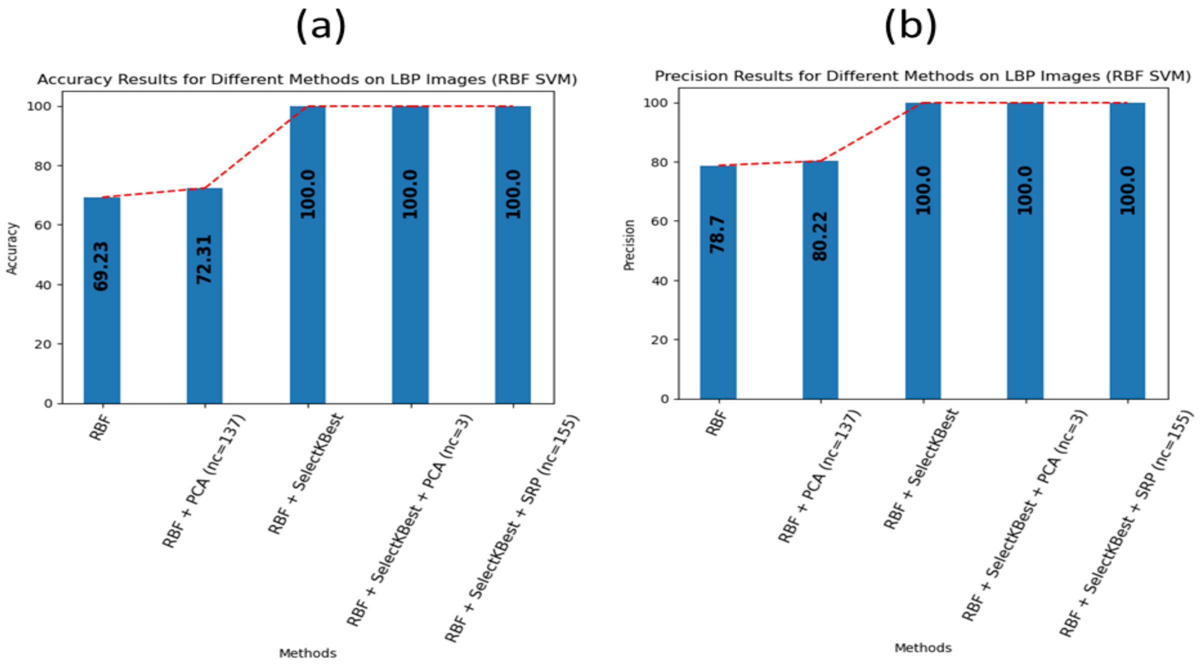


Figure V.22. (a) Accuracy, (b) Precision results on images with LBP using different methods with RBF SVM

5.4.3.3 ROI images with LBP

- **KNN**

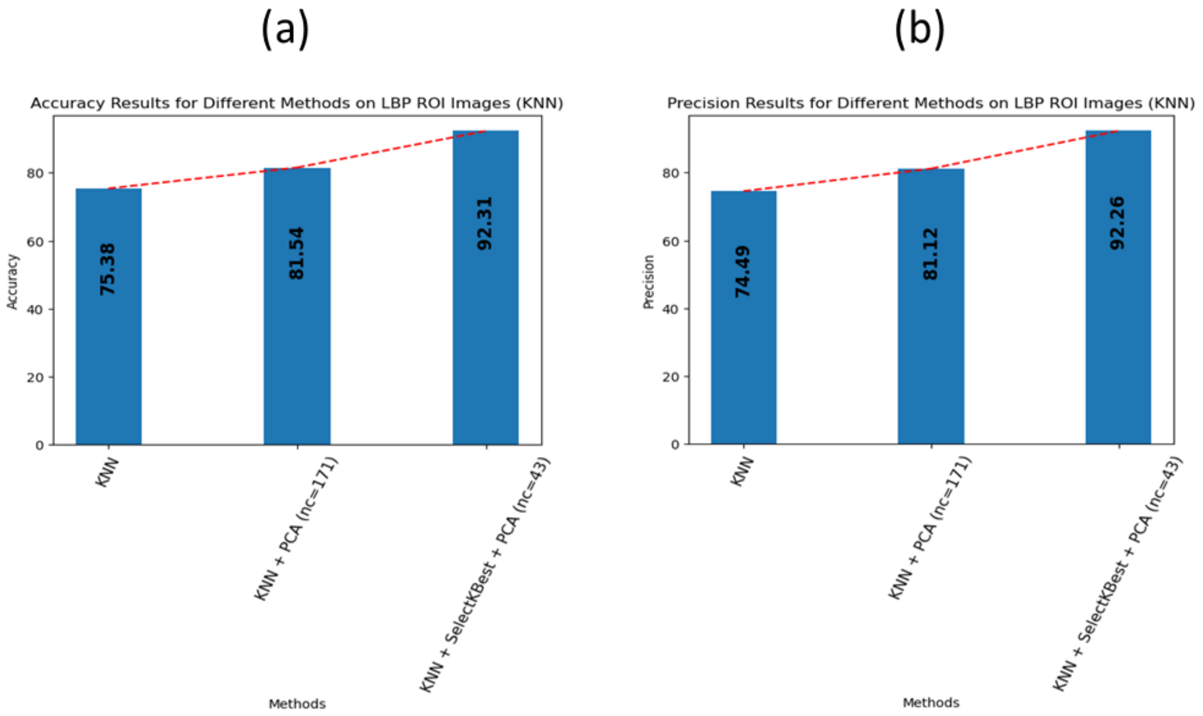


Figure V.23. (a) Accuracy, (b) Precision results on images with LBP using different methods with KNN

- **SVM**

I. Sigmoid kernel

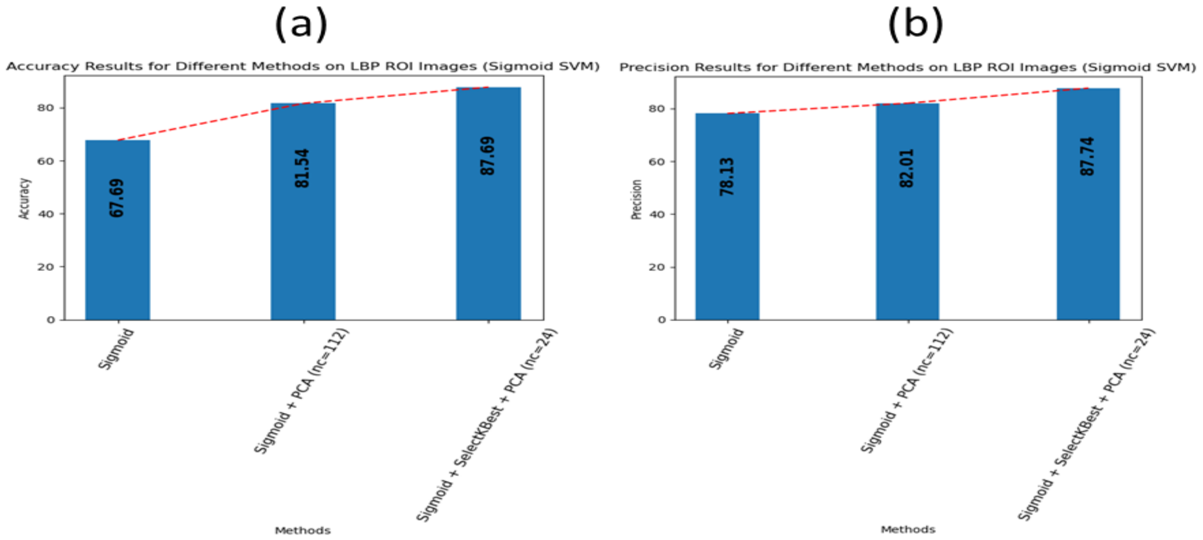


Figure V.24. (a) Accuracy, (b) Precision results on ROI images with LBP using different methods with Sigmoid SVM

II. Polynomial kernel

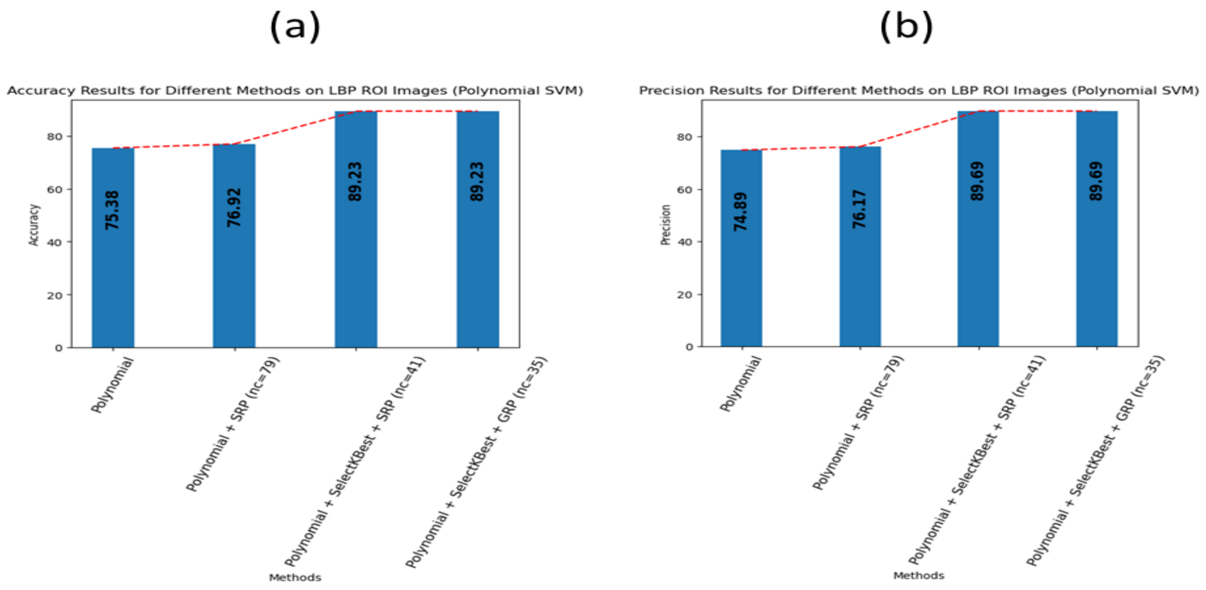


Figure V.25. (a) Accuracy, (b) Precision results on ROI images with LBP using different methods with Polynomial SVM

III. RBF kernel

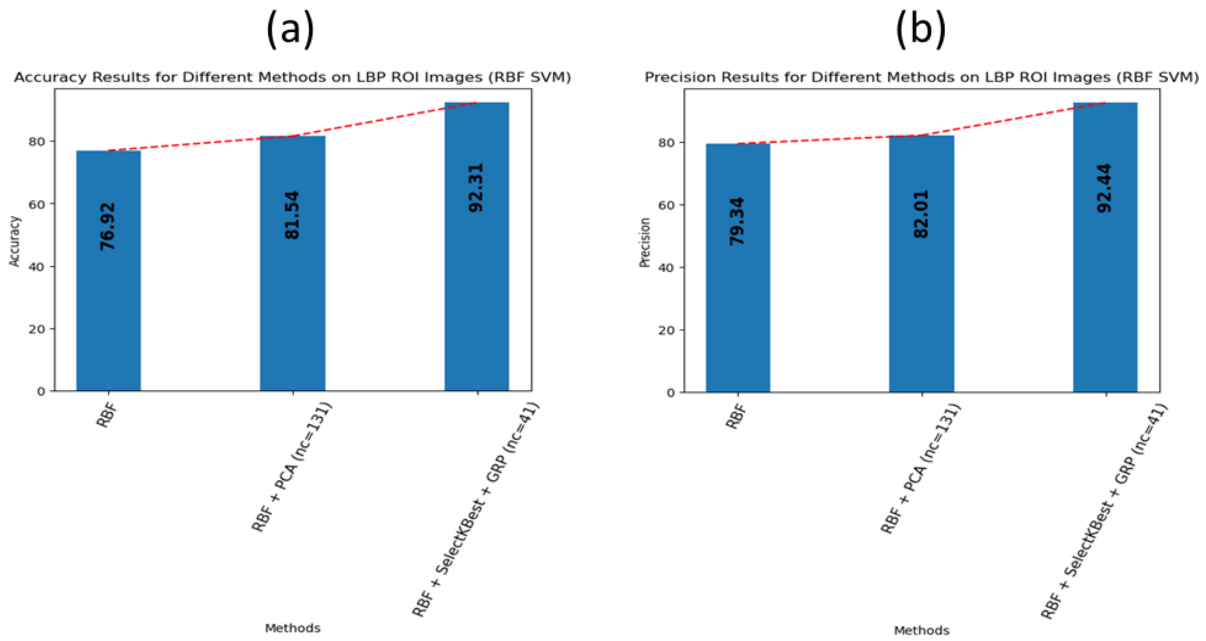


Figure V.26. (a) Accuracy, (b) Precision results on ROI images with LBP using different methods with RBF SVM

5.4.4 Best results

This section aims to present the best accuracy achieved by various methods and their corresponding precision values. The focus is on identifying the methods that yield the highest accuracy, which indicates the ability of the model to correctly predict whether an image is normal or abnormal.

5.4.4.1 Original images

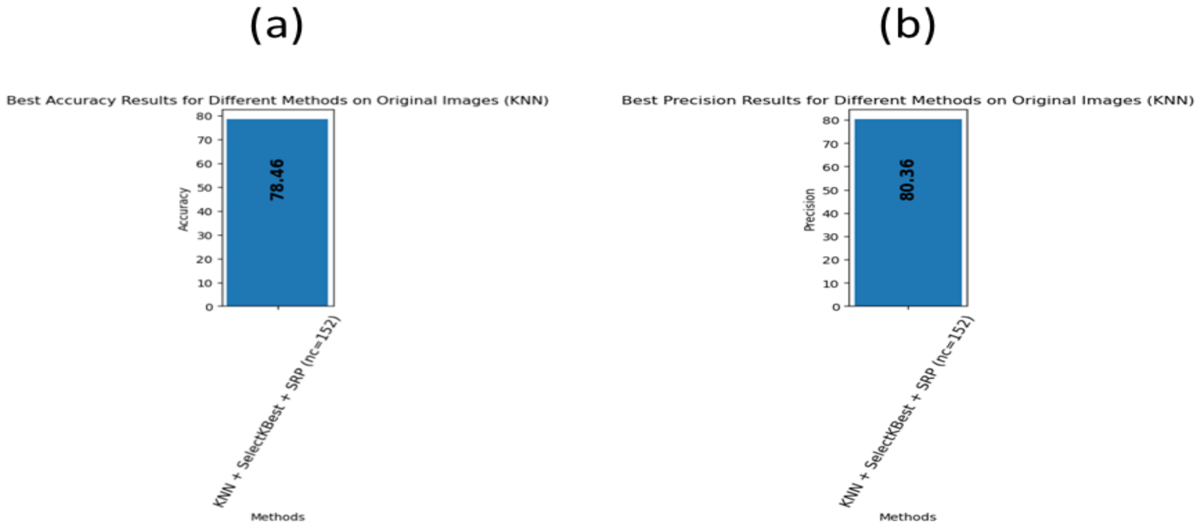


Figure V.27. Best (a) accuracy, (b) according precision results on original images with different methods using different classifiers (KNN, SVM)

5.4.4.2 Images with LBP

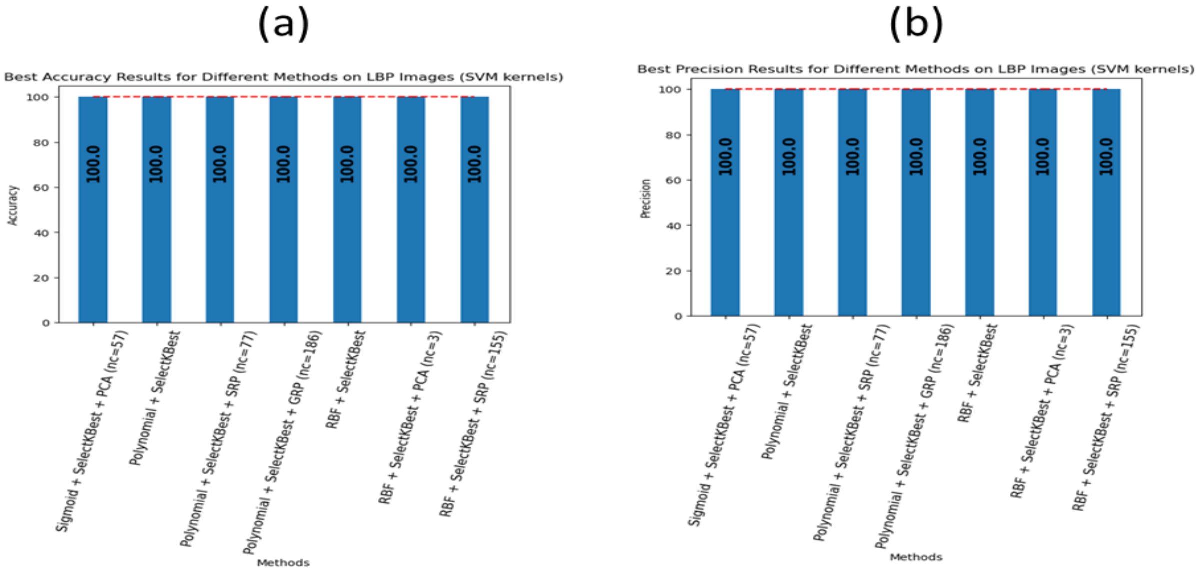


Figure V.28. Best (a) accuracy, (b) according precision results on LBP images with different methods using different classifiers (KNN, SVM)

5.4.4.3 ROI images with LBP

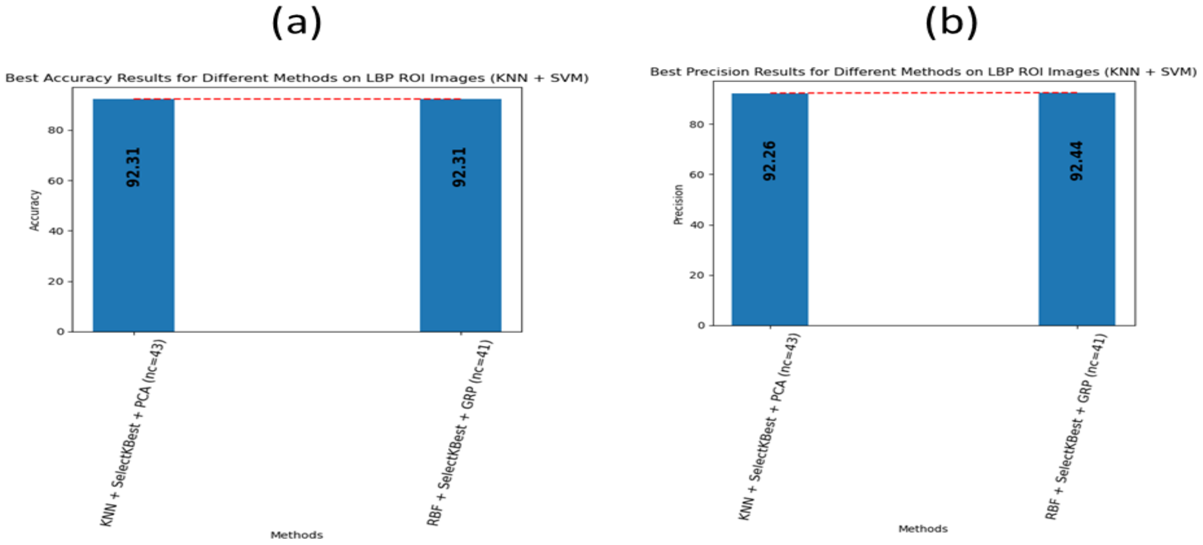


Figure V.29. Best (a) accuracy, (b) according precision results on LBP ROI images with different methods using different classifiers (KNN, SVM)

5.4.5 Predicted examples

In order to evaluate the performance of the models and assess their ability to achieve 100% accuracy, we conducted predictions on a set of images.

5.4.5.1 Polynomial kernel with SelectKBest on images with LBP

Table V.9. Predicted examples with Polynomial kernel combined with SelectKBest on images with LBP

Image	Prediction	Actual class
mdb002	Abnormal	Abnormal
mdb009	Normal	Normal
mdb237	Abnormal	Abnormal

5.4.5.2 RBF kernel with SelectKBest on images with LBP

Table V.10. Predicted examples with RBF kernel combined with SelectKBest on images with LBP

Image	Prediction	Actual class
mdb046	Normal	Normal
mdb001	Abnormal	Abnormal
mdb322	Normal	Normal

For the models (RBF + SelectKBest + (PCA or SRP)) and (Polynomial + (SRP or GRP)) and (Sigmoid + PCA) got overfit. These models are achieving 100% accuracy on the training data but are overfitting when making predictions on new, unseen data, it indicates that these models have memorized the training data instead of learning generalizable patterns.

5.4.5.3 RBF kernel with SelectKBest and PCA on images with LBP

Table V.11. All false predicted examples with RBF kernel combined with SelectKBest and PCA on images with LBP

Image	Prediction	Actual class
mdb072	Normal	Abnormal
mdb104	Normal	Abnormal
mdb178	Normal	Abnormal
mdb179	Normal	Abnormal
mdb213	Normal	Abnormal
mdb290	Normal	Abnormal

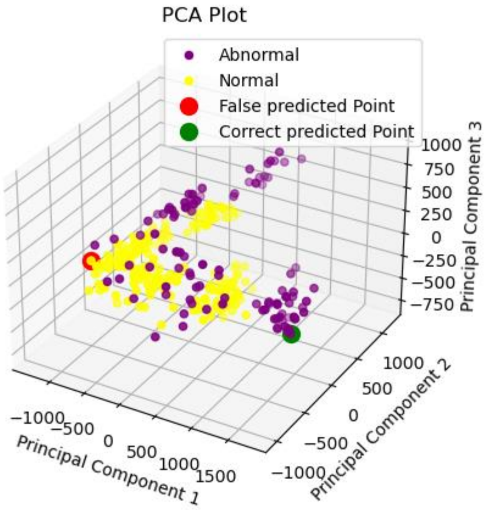


Figure V.30. 3D plot for the Model of RBF kernel combined with SelectKBest and PCA for images with LBP

5.4.5.4 RBF kernel with SelectKBest and SRP on images with LBP

Table V.12. All false predicted examples with RBF kernel combined with SelectKBest and SRP on images with LBP

Image	Prediction	Actual class
mdb104	Normal	Abnormal
mdb171	Normal	Abnormal
mdb178	Normal	Abnormal
mdb179	Normal	Abnormal
mdb285	Abnormal	Normal
mdb290	Normal	Abnormal

5.4.5.5 Polynomial kernel with SelectKBest and SRP on images with LBP

Table V.13. All false predicted examples with Polynomial kernel combined with SelectKBest and SRP on images with LBP

Image	Prediction	Actual class
mdb002	Normal	Abnormal
mdb032	Normal	Abnormal
mdb057	Abnormal	Normal
mdb077	Abnormal	Normal
mdb100	Abnormal	Normal
mdb107	Normal	Abnormal
mdb127	Normal	Abnormal
mdb147	Abnormal	Normal
mdb160	Normal	Abnormal
mdb178	Normal	Abnormal
mdb179	Normal	Abnormal
mdb189	Abnormal	Normal
mdb213	Normal	Abnormal
mdb227	Normal	Abnormal
mdb233	Normal	Abnormal
mdb279	Abnormal	Normal

5.4.5.6 Polynomial kernel with SelectKBest and GRP on images with LBP

Table V.14. All false predicted examples with Polynomial kernel combined with SelectKBest and GRP on images with LBP

Image	Prediction	Actual class
mdb072	Normal	Abnormal
mdb173	Abnormal	Normal
mdb178	Normal	Abnormal
mdb179	Normal	Abnormal
mdb213	Normal	Abnormal

5.4.5.7 *Sigmoid kernel with SelectKBest and PCA on images with LBP*

Table V.15. All false predicted examples with Sigmoid kernel combined with SelectKBest and PCA on images with LBP

Image	Prediction	Actual class
mdb025	Normal	Abnormal
mdb072	Normal	Abnormal
mdb090	Normal	Abnormal
mdb095	Normal	Abnormal
mdb104	Normal	Abnormal
mdb134	Normal	Abnormal
mdb171	Normal	Abnormal
mdb173	Abnormal	Normal
mdb178	Normal	Abnormal
mdb197	Abnormal	Normal
mdb233	Normal	Abnormal
mdb271	Normal	Abnormal
mdb295	Abnormal	Normal

5.5 Discussion

5.5.1 Original images

According to the results from table 5.3. We can discuss about the performance of the two classifiers: KNN and SVM with different methods.

- **KNN:** The results obtained using the KNN classifier were as follows: accuracy (75.38%), precision (75.3%), recall (75.38%), F1-score (71.56%), sensitivity (95.56%), specificity (30%) and average time (2.45s).

  When applying dimension reduction techniques, the accuracy, precision, recall, F1-score, sensitivity, and specificity generally improved, except for LDA that showed a decrease in performance. Among the dimension reduction techniques, the best result was achieved using SRP, which increased the metrics of accuracy, precision, recall, F1-

score and specificity by 3.08%, 3.43%, 3.08%, 4.39%, 10% and respectively, while sensitivity remained the same and the average time decreased by 0.35s.

Next, when incorporating feature selection with the SelectKBest technique, there was an increase in precision, sensitivity and average time from 75.38%, 95.56% and 2.45s to 77.27%, 97.78% and 2.90s respectively. However, there was a decrease in F1-score

Finally, combining feature selection and dimension reduction techniques, the best result was obtained using SRP, which further improved the metrics (accuracy, precision, recall, F1-score, sensitivity, and specificity) of the KNN classifier by 3.08%, 5.06%, 3.08%, 3.55%, 2.22%, and 5% respectively while there was a little increase in average time by 0.10s.

These results indicate that the application of dimension reduction techniques and feature selection can generally improve the performance of the KNN classifier on the original images. The SRP technique consistently yielded the best results in terms of various metrics, while LDA showed a decrease in performance.

- **SVM:** When using the SVM classifier with the three kernels (sigmoid, polynomial, radial basis function), applying dimension reduction techniques generally improved the results for all kernels (every kernel with a specific technique and not all of them) except for LDA, which showed a decrease in performance. Focusing on the best results, they were achieved with the sigmoid kernel.
- **Sigmoid kernel:** The initial metrics were: accuracy (75.38%), precision (75.30%), recall (75.38%), F1-score (71.56%), sensitivity (95.56%), specificity (30%) and average time (45.43s).

When incorporating dimension reduction techniques, there was an improvement in the metrics values by 1.54%, 3.59%, 1.54%, 1.26% and 2.22% respectively with a decrease in average time by 43.69s. Only the specificity has remained the same. These results were achieved by the SRP technique.

Next, applying SelectKBest alone resulted in a decrease in the metrics results by 9.23%, 10.07%, 9.23%, 5.93% 21.65s, and an increase in specificity by 10%.

Finally, when combining SelectKBest with dimension reduction techniques (SRP), the overall results were: accuracy (75.38%), precision (81.84%), recall (75.38%), F1-score (69.04%), sensitivity (100%), specificity (20%) and average time (2.36s). However, when comparing these combined results with the ones of the SRP technique only, it showed that they were not better.

5.5.2 Images with LBP

The results from table 5.4 that represents the performance of the two classifiers: KNN and SVM with different methods are discussed.

- **KNN:** The initial results were: accuracy (61.54%), precision (46.15%), recall (61.54%), F1-score (52.75%), sensitivity (88.89%), specificity (0%), and average time (2.19 seconds).

When applying dimension reduction techniques, all methods showed improvement, with GRP yielding the best results. The metrics increased by 16.92%, 37.42%, 16.92%, 21.36%, 11.11%, 30%, and 2.15 seconds respectively compared to the KNN without dimension reduction techniques. Despite the fact that dimension reduction improved the classification results, it also introduced a trade-off in terms of computational efficiency.

Using the SelectKBest technique alone resulted in improved metrics with the following results: accuracy (78.46%), precision (83.57%), recall (78.46%), F1-score (74.11%), sensitivity (100%), specificity (30%), and average time (0.01 seconds).

Combining SelectKBest with dimension reduction techniques, PCA showed the best results. Compared to KNN only, the metrics improved by 35.38%, 50.9%, 35.38%, 44.13%, 11.11%, 90%, and the average time decreased by 2.17 seconds. Compared to KNN with SelectKBest only, the metrics increased by 18.46%, 13.48%, 18.46%, 22.77%, 0%, 60%, and 0.01 seconds.

In summary, for the KNN classifier, incorporating dimension reduction techniques (particularly GRP) and SelectKBest resulted in improved metrics. The combination of **SelectKBest** and **PCA** provided the best results (**96.92%, 97.05%, 96.92%, 96.88%, 100%, 90%, 0.02s**), significantly enhancing the performance and reducing the execution time compared to the initial KNN with LBP.

- **SVM**

- **Sigmoid kernel:** When using the SVM classifier with the sigmoid kernel, the results showed an accuracy of 69.23%, precision of 78.70%, recall of 69.23%, F1-score of 56.64%, sensitivity of 100%, specificity of 0%, and an average execution time of 44.97 seconds.

Applying dimension reduction techniques improved the results, particularly with PCA and GRP. SRP had minimal impact, while LDA resulted in decreased performance. The best improvement was observed with PCA, increasing the metrics by 1.54%, 0.75%, 1.54%, 3.45%, and 5% for accuracy, precision, recall, F1-score, and specificity respectively. Additionally, the average execution time was reduced by 11.08 seconds.

Using SelectKBest with the sigmoid kernel resulted in similar performance, with accuracy, precision, recall, F1-score, sensitivity, specificity, and average execution time of 69.23%, 78.70%, 69.23%, 56.64%, 100%, 0%, and 0.03 seconds respectively.

Finally, when combining SelectKBest with dimension reduction techniques, PCA yielded the best results. The metrics increased by 30.77%, 21.30%, 30.77%, 43.36%, and 100% for accuracy, precision, recall, F1-score, and specificity respectively, compared to the sigmoid kernel without any dimension reduction or feature selection. Furthermore, the average execution time was reduced by 44.94 seconds. Similar results were observed when comparing with the sigmoid kernel + SelectKBest, with the only difference being the average execution time remaining the same.

Overall, applying dimension reduction techniques and feature selection improved the performance of the sigmoid kernel, with **SelectKBest + PCA** consistently yielding the best results **(100%, 100%, 100%, 100%, 100%, 100%, 0.03s)** across the evaluated metrics.

- **Polynomial kernel:** When using the SVM classifier with the polynomial kernel, the results showed an accuracy of 69.23%, precision of 78.70%, recall of 69.23%, F1-score of 56.64%, sensitivity of 100%, specificity of 0%, and an average execution time of 59.34 seconds.

Applying dimension reduction techniques led to improved performance, except for LDA which remained the same. SRP yielded the best results, increasing the metrics (accuracy, precision, recall, F1-score, sensitivity, specificity) by 3.08%, 1.52%, 3.08%, 6.65%, 0%, and 10% respectively, while reducing the average time by 57.31 seconds.

Using SelectKBest also resulted in improved metrics, with an increase of 30.77%, 21.30%, 30.77%, 43.36%, 0%, and 100% respectively, and a decrease in average execution time by 59.34 seconds.

When combining SelectKBest with dimension reduction techniques, SRP and GRP showed similar results to SelectKBest, with slight improvements in average execution time. PCA and LDA yielded better results compared to the polynomial kernel alone, but with a slight decrease compared to the polynomial kernel with SelectKBest or SelectKBest combined with SRP/GRP.

Overall, applying **SelectKBest** only or with dimension reduction techniques improved the performance of the polynomial kernel, with different techniques yielding varying levels of enhancement in accuracy, precision, recall, F1-score, sensitivity, specificity, and average execution time. Combining **SelectKBest with SRP or GRP** gave the best performance with 100% for all metrics and an average time of 0.3 seconds.

- **Radial Basis Function (RBF) kernel:** The initial results of the RBF kernel showed an accuracy of 69.23%, precision of 78.70%, recall of 69.23%, F1-score of 56.64%, sensitivity of 100%, specificity of 0% and an average time of 126.37 seconds.

  When dimension reduction techniques were applied, the performance improved, except for LDA which remained unchanged. Notably, PCA yielded the most significant enhancements, increasing accuracy by 3.08%, precision by 1.52%, recall by 3.08%, F1-score by 6.65%, and specificity by 10%. Moreover, it reduced the average execution time by 81.95 seconds.

  Using the SelectKBest feature selection method further elevated the results, with a notable increase of 30.77% in accuracy, 21.30% in precision, 30.77% in recall, 43.36% in F1-score, and 100% in specificity. Additionally, it reduced the average execution time by 126.33 seconds.

  When combined with dimension reduction techniques, both PCA and SRP produced comparable results to SelectKBest, with minor improvements in average execution time by 0.02 seconds. Furthermore, GRP and LDA methods exhibited superior results compared to the RBF kernel alone, although there was a slight decrease when compared to SelectKBest or SelectKBest in conjunction with PCA and SRP.

  In summary, employing SelectKBest alone or dimension reduction techniques, particularly PCA and SRP, in conjunction with SelectKBest enhanced the performance of the RBF kernel, resulting in improved metrics (100% for all), while reducing the average execution time.

5.5.3 ROI images with LBP

From the results of table 5.5. We can discuss about the various techniques used which are combined with two classifiers: KNN and SVM.

- **KNN:** The initial results using KNN showed an accuracy of 75.38%, precision of 74.49%, recall of 75.38%, F1-score of 74.62%, sensitivity of 86.36%, specificity of 52.38%, and the average time was 0.04 seconds.

Applying dimension reduction techniques yielded improvements, with PCA and SRP increasing the results while GRP and LDA decreased them. Notably, PCA provided the best results, enhancing accuracy by 6.16%, precision by 6.63%, recall by 6.16%, F1-score by 6.35%, sensitivity by 4.55%, specificity by 9.52%, and the average time by 0.64 seconds.

Using the SelectKBest feature selection technique led to further enhancements, resulting in an accuracy, precision, recall, and F1-score of 87.69%. The sensitivity improved to 90.91%, specificity to 80.95%, and the average execution time decreased by 0.02 seconds. Compared to the KNN without feature selection, the metrics increased by 12.31%, 13.20%, 12.31%, and 13.07%, 4.55% and 28.57% respectively.

Combining SelectKBest with dimension reduction techniques further improved the results compared to KNN only, but showed mixed effects compared to KNN + SelectKBest. PCA and GRP yielded increased metrics, while LDA decreased them. SRP yielded similar results. PCA achieved the best performance, with improvements of 4.62%, 4.57%, 4.62%, and 4.57%, 4.54, 4.76 for accuracy, precision, recall, F1-score, sensitivity and specificity respectively. The average execution time remained unchanged.

In summary, applying dimension reduction techniques, especially PCA, in combination with SelectKBest resulted in improved classification results for ROI images with LBP, enhancing accuracy, precision, recall, and F1-score while maintaining or reducing the average execution time.

- **SVM:** the performance of each kernel is analyzed individually:

➢ Sigmoid kernel:

○ Applying dimension reduction techniques resulted in increased results overall, except for LDA which showed a decrease.

○ SelectKBest with the sigmoid kernel did not significantly affect the results, except for a decrease in average time.

○ Combining SelectKBest with dimension reduction techniques yielded improved results across all metrics.

➢ Polynomial kernel:

○ When using dimension reduction techniques, the results improved only with SRP, while they decreased for the other techniques.

○ SelectKBest had mixed effects, increasing some metrics while decreasing others.

○ Combining SelectKBest with dimension reduction techniques resulted in improved results across all techniques.

○

- RBF kernel:
  - Initially, the results were 76.92% for accuracy, 79.34% for precision, 76.92% for recall, 73.24% for F1-score, 97.73% for sensitivity, 33.33% for specificity, and 1.01 seconds for average time.
  - Applying dimension reduction techniques resulted in decreased results for GRP and increased results for the other techniques. PCA showed the best improvement, increasing accuracy by 4.62%, precision by 2.67%, recall by 4.62%, F1-score by 6.90%, and specificity by 19.05%. However, sensitivity decreased by 2.28%, and the average time decreased by 0.36 seconds.
  - SelectKBest improved several metrics, with increases of 7.70% for accuracy, 5.09% for precision, 7.70% for recall, 10.9% for F1-score, and 33.34% for specificity. However, sensitivity decreased by 4.55%, and the average time decreased by 0.99 seconds. The results became 84.62% for accuracy, 84.43% for precision, 84.62% for recall, 84.14% for F1-score, 93.18% for sensitivity, 66.67% for specificity, and 0.02 seconds for average time.
  - Combining SelectKBest with dimension reduction techniques resulted in decreased results for LDA and improved results for the other techniques. GRP showed the best improvement, increasing accuracy by 7.69%, precision by 8.01%, recall by 7.69%, F1-score by 8%, and specificity by 4.55%. The average time remained the same.

In summary, the performance of the SVM classifier varied depending on the kernel used. Dimension reduction techniques generally improved the results, except for specific cases such as LDA with the sigmoid kernel. SelectKBest had mixed effects, improving some metrics while decreasing others. Combining SelectKBest with dimension reduction techniques often led to improved results. Overall, the best results were observed with the RBF kernel using **GRP** in combination with **SelectKBest (92.31%, 92.44%, 92.31%, 92.14%, 97.73%, 80.95%, 0.02s)**, with improvements in various metrics while maintaining or decreasing the average time.

5.5.4 Discussion summary

In order to summarize what was discussed before:

- For the original images, according to the results shown in figure V.16, the best performance was achieved by combining SelectKBest for feature selection, SRP for dimension reduction, and the KNN classifier. This combination resulted in an accuracy of 78.46% and a precision of 80.36%. It can be concluded that the combination of feature selection and dimension reduction can enhance the classification performance. This is evident from the improved accuracy, precision, and average execution time observed when using these techniques together. However, it should be noted that the absence of feature extraction might have affected the performance of the feature selection technique. The selected features may not have been the most relevant ones for the classification task, leading to limited improvements. This what we’re going to explore in the next section.
- For the images with features extracted using LBP, from figure V.17, it’s clear that the SVM classifier outperformed KNN, achieving a maximum performance of 100% for all computed metrics. All the kernel functions used in SVM showed maximum performance when combined with feature selection (SelectKBest) or combined with both feature selection and dimension reduction techniques (PCA, SRP, and GRP). The sigmoid kernel was the exception, requiring the addition of a dimension reduction technique to achieve maximum performance. Among the dimension reduction techniques, LDA performed the worst. This could be attributed to LDA's focus on separating classes, which may not be as effective in a binary classification scenario. LDA may perform better in multi-class classification tasks. To choose among the combinations that achieved 100% performance, considering the average time can be a deciding factor. This factor alone may not be sufficient to choose the best model. While achieving 100% performance on the training data is desirable, it does not guarantee that the model will perform equally well on unseen data. To ensure the reliability of the chosen model and minimize errors, it is essential to evaluate its performance on unseen data. By making predictions on a separate validation or test set, we can assess how well the model generalizes to new examples. This evaluation provides a more realistic estimate of the model's performance and helps identify any potential overfitting issues. According to the tables from V.6 to V.12, the options to consider are polynomial kernel + SelectKBest and RBF kernel + SelectKBest because the other models showed an

overfit with false predictions on some images. These results highlight the importance of combining feature extraction, feature selection, and dimension reduction techniques. Feature extraction extracts relevant features, feature selection selects the most informative ones, and dimension reduction retains the most informative features for improved machine learning algorithm performance.

In conclusion, the combination of feature extraction, feature selection, and dimension reduction techniques can significantly enhance the performance of machine learning algorithms, as demonstrated in the achieved results. The appropriate selection and combination of these techniques can lead to optimal classification performance and efficient use of computational resources.

- For the ROI images with features extracted using LBP, according to the results of the figure V.18, the SVM and KNN classifiers performed similarly in terms of accuracy, with a slight advantage for the SVM classifier in precision. The accuracy and precision results were (92.31% and 92.26%) for the KNN classifier and (92.31% and 92.44%) for the SVM classifier. Considering only accuracy and precision, the SVM classifier with the RBF kernel + SelectKBest + GRP combination gave the best result, with a slightly higher precision score. For other metrics and average times, the performance of KNN and SVM classifiers was comparable. Therefore, the choice between these two classifiers depends on the specific requirements and priorities of the application.

  Overall, the combination of dimension reduction, feature selection, and feature extraction techniques is crucial for improving classification performance and building accurate prediction models.

5.6 Conclusion

In conclusion, the results and discussions presented in this chapter provide valuable insights into the performance and effectiveness of various classification approaches and techniques. The evaluation of the classification task has allowed us to assess the strengths and limitations of different methods, enabling us to make informed decisions and improvements in the modeling

Throughout the evaluation, we have observed the impact of feature extraction, feature selection, and dimension reduction techniques on the classification performance. It became evident that the combination of these techniques can significantly enhance the accuracy, precision, and the overall performance of the classification models. The use of feature extraction techniques such as LBP has proven beneficial in capturing relevant information from the images, while feature selection methods like SelectKBest have aided in selecting the most informative features for classification.

Moreover, dimension reduction techniques such as PCA, SRP, and GRP have played a crucial role in reducing the dimensionality of the feature space, improving computational efficiency, and in some cases, enhancing the classification performance. However, it is important to note that the choice of dimension reduction technique should be carefully considered, as different techniques such as LDA in our case may yield varying results depending on the specific dataset and classification problem. When working with large datasets, it is important to consider the computational efficiency of dimension reduction techniques. While PCA has demonstrated its effectiveness in reducing the dimensionality of feature space and capturing informative features, it may not be the most suitable choice for large datasets due to its memory requirements and time complexity. Instead, techniques such as Sparse Random Projection (SRP) and Gaussian Random Projection (GRP) offer attractive alternatives in terms of computational efficiency. These techniques provide fast and scalable solutions for dimension reduction, making them well-suited for both large and small datasets.

Furthermore, the evaluation of different kernel functions in the SVM classifier has revealed their varying effectiveness in capturing the underlying patterns in the data. While some kernels, such as the RBF kernel, have shown remarkable performance across multiple metrics especially with a reduced dimension which means that RBF can perform well for small datasets.

Overall, the evaluation of the classification task has emphasized the significance of considering a combination of feature extraction, feature selection, and dimension reduction techniques to improve the performance of classification models. By carefully selecting and integrating these techniques, it is possible to achieve higher accuracy, precision, and overall reliability in the classification results.

General Conclusion

In this study, we implement a CAD system and investigate effects of dimension reduction for classifying mammograms. **The results demonstrate the effectiveness of dimensionality reduction techniques, especially when combined with SelectKBest, in improving the classification accuracy for the detection of abnormalities in mammograms.** The best accuracy of 100% was achieved when combining SelectKBest with specific kernels of SVM and dimension reduction techniques which wasn't achieved without using these techniques. This study has several important implications. First, it suggests that dimensionality reduction techniques can be a valuable tool for improving the accuracy of CAD systems for breast cancer detection. Second, it shows that SelectKBest can be an effective feature selection technique for this task. Third, it provides evidence that the combination of dimensionality reduction and feature selection can lead to significant accuracy improvements especially PCA and RP. Linear Discriminant Analysis (LDA) did not outperform the other techniques for this specific task. However, it is important to highlight that the performance of LDA may vary depending on the nature of the task, the way it is used and the characteristics of the dataset. LDA has shown promising results in various applications and may be more suitable for other classification tasks. It is important to acknowledge the limitations of this study. The research was conducted on MIAS images which represent a small dataset, and the performance of the system should be further evaluated on larger, more diverse datasets to establish its robustness and generalizability. Additionally, the study focused on a specific CAD system and the performance may vary when applied to different systems or imaging modalities. Moving forward, future research should aim to validate the findings on larger datasets and explore the performance of the system across different populations and imaging techniques. Additionally, the integration of other advanced techniques, such as deep learning algorithms or other dimension reduction techniques with different classifiers could be considered to further improve the accuracy of breast cancer detection. In conclusion, this study contributes to the field of breast cancer detection by demonstrating the efficacy of dimensionality reduction techniques, particularly when combined with SelectKBest, in improving the classification accuracy of mammograms. The findings confirm the potential of these techniques to enhance early detection, ultimately leading to better patient outcomes. However, further experiments and validation on larger datasets are essential to solidify the results and facilitate the translation of these findings into clinical practice.

Bibliography

[7] Y. S. Sun, Z. Zhao, Z. N. Yang, F. Xu, H. J. Lu, Z. Y. Zhu, ... & et H. P. Zhu, «Risk factors and preventions of breast cancer,» *International journal of biological sciences, 13(11),* p. 1387, 2017.

[17] D. Simos, C. Catley, C. van Walraven, A. Arnaout, C. M. Booth, M. McInnes, ... et M. Clemons, «Imaging for distant metastases in women with early-stage breast cancer: a population-based cohort study,» *Cmaj, 187(12),* pp. E387-E397, 2015.

[41] W. R. E et G. R. & C, Digital image processing, 2008.

[50] A. AlQoud et M. A. Jaffar , «Hybrid gabor based local binary patterns texture features for classification of breast mammograms,» *Int. J. Comput. Sci. Netw. Secur.(IJCSNS), 16(4),* p. 16, 2016.

[53] L. Armi et S. Fekri-Ershad, «Texture image analysis and texture classification methods-A review,» *arXiv preprint arXiv:1904.06554,* 2019.

[54] A. Elmoufidi, K. El Fahssi, S. Jai-Andaloussi, N. Madrane et A. Sekkaki, «Detection of regions of interest's in mammograms by using local binary pattern, dynamic k-means algorithm and gray level co-occurrence matrix,» *In 2014 International Conference on Next Generation Networks and Services (NGNS),* pp. (pp. 118-123). IEEE, 2014, May.

[55] A. Tosin, A. Morufat, O. Omotayo, W. Bolanle, O. Olusayo et O. Olatunde, «Curvelet transform-local binary pattern feature extraction technique for mass detection and classification in digital mammogram,» *Current Journal of Applied Science and Technology, 28(3),* pp. 1-15, 2018.

[56] J. Brownlee, «How to choose a feature selection method for machine learning,» *Machine Learning Mastery,* p. 10, 2019.

[57] A. L. Blum et P. & Langley, «Selection of relevant features and examples in machine learning,» *Artificial intelligence, 97(1-2),* pp. 245-271, 1997.

[58] J. Li, K. Cheng, S. Wang, F. Morstatter, R. P. Trevino, J. Tang et H. Liu, «Feature selection: A data perspective,» *ACM computing surveys (CSUR), 50(6),* pp. 1-45, 2017.

[60] M. A. Carreira-Perpinán, «A review of dimension reduction techniques,» *Department of Computer Science. University of Sheffield. Tech. Rep. CS-96-09, 9,* pp. 1-69, 1997.

[62] P. S. Hajare et V. V. Dixit, «Breast tissue classification using gabor filter, PCA and support vector machine,» *International Journal of advancement in electronics and computer engineering (IJAECE), 1(4),* pp. 116-119, 2012.

[64] J. Maudes, J. J. Rodríguez, C. García-Osorio et C. Pardo, «Random projections for linear SVM ensembles,» *Applied Intelligence, 34,* pp. 347-359, 2011.

[65] E. Bingham et H. Mannila, «Random projection in dimensionality reduction: applications to image and text data.,» *In Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining* , pp. (pp. 245-250), 2001, August.

[67] A. Mohammed, A. Ahmed, W. Mohammed, G. K. Viju et M. Taha, «Mammogram images classification using linear discriminant analysis,» *Int. Res. J. Eng. Technol (IRJET), 7,* p. 6, 2020.

[84] P. Sonar, U. Bhosle et C. Choudhury, «Mammography classification using modified hybrid SVM-KNN,» *In 2017 international conference on signal processing and communication (ICSPC),* pp. (pp. 305-311). IEEE, 2017, July.

[90] S. Suthaharan et S. . Suthaharan, «Support vector machine,» *Machine learning models and algorithms for big data classification: thinking with examples for effective learning,* pp. 207-235, 2016.

Webography

[1] «What is Breast Cancer?,» [En ligne]. Available: https://www.komen.org/breast-cancer/facts-statistics/what-is-breast-cancer/. [Consulted on 17 5 2023].

[2] «Anatomy of the Breast,» [En ligne]. Available: https://www.komen.org/breast-cancer/facts-statistics/what-is-breast-cancer/the-breast-anatomy/. [Consulted on 17 5 2023].

[3] «Breast Cancer,» [En ligne]. Available: https://my.clevelandclinic.org/health/diseases/3986-breast-cancer. [Consulted on 17 5 2023].

[4] «What Is Breast Cancer?,» [En ligne]. Available: https://www.cdc.gov/cancer/breast/basic_info/what-is-breast-cancer.htm. [Consulted on 17 5 2023].

[5] «Male Breast Cancer,» [En ligne]. Available: https://www.nationalbreastcancer.org/male-breast-cancer/. [Consulted on 17 5 2023].

[6] «What Are the Risk Factors for Breast Cancer?,» [En ligne]. Available: https://www.cdc.gov/cancer/breast/basic_info/risk_factors.htm. [Consulted on 17 5 2023].

[8] «How Is Breast Cancer Diagnosed?,» [En ligne]. Available: https://www.cdc.gov/cancer/breast/basic_info/diagnosis.htm. [Consulted on 17 5 2023].

[9] «Diagnosis,» [En ligne]. Available: https://www.nationalbreastcancer.org/breast-cancer-diagnosis/. [Consulted on 17 5 2023].

[10] «Diagnostic Mammogram,» [En ligne]. Available: https://www.nationalbreastcancer.org/diagnostic-mammogram/. [Consulted on 17 5 2023].

[11] «Ultrasound,» [En ligne]. Available: https://www.nationalbreastcancer.org/breast-ultrasound/. [Consulted on 17 5 2023].

[12] «MRI,» [En ligne]. Available: https://www.nationalbreastcancer.org/breast-mri/. [Consulted on 17 5 2023].

[13] «Breast Cancer Diagnosis,» [En ligne]. Available: https://www.komen.org/breast-cancer/diagnosis/. [Consulted on 17 5 2023].

[14] «Breast Biopsy,» [En ligne]. Available: https://www.nationalbreastcancer.org/breast-cancer-biopsy/. [Consulted on 17 5 2023].

[15] «Breast Positron Emission Tomography,» [En ligne]. Available: https://emedicine.medscape.com/article/2109054-overview. [Consulted on 17 5 2023].

[16] «Breast Cancer Stages,» [En ligne]. Available: https://www.nationalbreastcancer.org/about-breast-cancer/breast-cancer-staging/. [Consulted on 17 5 2023].

[18] «Stages 0 & 1 Breast Cancer Overview,» [En ligne]. Available: https://www.nationalbreastcancer.org/breast-cancer-stage-0-and-stage-1/. [Consulted on 17 5 2023].

[19] «Stage 2 (II) And Stage 2A (IIA) Breast Cancer Overview,» [En ligne]. Available: https://www.nationalbreastcancer.org/breast-cancer-stage-2/. [Consulted on 17 5 2023].

[20] «Stage 3 (III) A, B, And C Breast Cancer Overview,» [En ligne]. Available: https://www.nationalbreastcancer.org/breast-cancer-stage-3/. [Consulted on 17 5 2023].

[21] «Stage 4 (IV) Breast Cancer,» [En ligne]. Available: https://www.nationalbreastcancer.org/breast-cancer-stage-4/. [Consulted on 17 5 2023].

[22] «Understanding the Stages of Breast Cancer,» [En ligne]. Available: https://www.healthline.com/health/breast-cancer/what-are-the-4-stages-of-breast-cancer. [Consulted on 20 5 2023].

[23] «Invasive Ductal Carcinoma (IDC),» [En ligne]. Available: https://www.nationalbreastcancer.org/invasive-ductal-carcinoma/. [Consulted on 17 5 2023].

[24] «Ductal Carcinoma In Situ (DCIS),» [En ligne]. Available: https://www.nationalbreastcancer.org/dcis/. [Consulted on 17 5 2023].

[25] «Breast cancer,» [En ligne]. Available: https://www.mayoclinic.org/diseases-conditions/breast-cancer/symptoms-causes/syc-20352470. [Consulted on 17 5 2023].

[26] «Lobular Carcinoma In Situ (LCIS),» [En ligne]. Available: https://www.nationalbreastcancer.org/lobular-carcinoma-in-situ/. [Consulted on 17 5 2023].

[27] «Lobular carcinoma in situ (LCIS),» [En ligne]. Available: https://www.mayoclinic.org/diseases-conditions/lobular-carcinoma-in-situ/symptoms-causes/syc-20374529. [Consulted on 17 5 2023].

[28] «Triple Negative Breast Cancer,» [En ligne]. Available: https://www.nationalbreastcancer.org/triple-negative-breast-cancer/. [Consulted on 17 5 2023].

[29] «Inflammatory Breast Cancer (IBC),» [En ligne]. Available: https://www.nationalbreastcancer.org/inflammatory-breast-cancer/. [Consulted on 17 5 2023].

[30] «Metastatic Breast Cancer,» [En ligne]. Available: https://www.nationalbreastcancer.org/metastatic-breast-cancer/. [Consulted on 17 5 2023].

[31] «Breast Tumors,» [En ligne]. Available: https://www.nationalbreastcancer.org/breast-tumors/. [Consulted on 17 5 2023].

[32] «How Is Breast Cancer Treated?,» [En ligne]. Available: https://www.cdc.gov/cancer/breast/basic_info/treatment.htm. [Consulted on 17 5 2023].

[33] «Treatment,» [En ligne]. Available: https://www.nationalbreastcancer.org/breast-cancer-treatment/. [Consulted on 17 5 2023].

[34] «Choosing Your Doctor,» [En ligne]. Available: https://www.nationalbreastcancer.org/breast-cancer-doctors/. [Consulted on 17 5 2023].

[35] «Surgery,» [En ligne]. Available: https://www.nationalbreastcancer.org/breast-cancer-surgery/. [Consulted on 17 5 2023].

[36] «Breast Cancer Treatment (PDQ®)–Patient Version,» [En ligne]. Available: https://www.cancer.gov/types/breast/patient/breast-treatment-pdq#_148. [Consulted on 20 5 2023].

[37] «Chemotherapy,» [En ligne]. Available: https://www.nationalbreastcancer.org/breast-cancer-chemotherapy/. [Consulted on 17 5 2023].

[38] «Types of Radiation Therapy for Breast Cancer,» [En ligne]. Available: https://www.verywellhealth.com/breast-cancer-radiation-methods-430554. [Consulted on 22 5 2023].

[39] «Hormone therapy,» [En ligne]. Available: https://www.cancerresearchuk.org/about-cancer/breast-cancer/treatment/hormone-therapy. [Consulted on 22 5 2023].

[40] «Targeted Therapy,» [En ligne]. Available: https://www.nationalbreastcancer.org/breast-cancer-targeted-therapy/. [Consulted on 17 5 2023].

[42] «Image Processing,» [En ligne]. Available: https://uotechnology.edu.iq/ce/lecture%202013n/4th%20Image%20Processing%20_Lectures/DIP_Lecture2.pdf. [Consulted on 2 6 2023].

[43] «The mini-MIAS database of mammograms,» [En ligne]. Available: http://peipa.essex.ac.uk/info/mias.html. [Consulted on 3 6 2023].

[44] «Types Of Digital Images,» [En ligne]. Available: https://blog.oureducation.in/types-of-digital-image/. [Consulted on 2 6 2023].

[45] «Noise in Digital Image Processing,» [En ligne]. Available: https://medium.com/image-vision/noise-in-digital-image-processing-55357c9fab71. [Consulted on 2 6 2023].

[46] «Intensity Histogram,» [En ligne]. Available: https://homepages.inf.ed.ac.uk/rbf/HIPR2/hipr_top.htm. [Consulted on 3 6 2023].

[47] «Introduction to Histogram Equalization for Digital Image Enhancement,» [En ligne]. Available: https://levelup.gitconnected.com/introduction-to-histogram-equalization-for-digital-image-enhancement-420696db9e43. [Consulted on 3 6 2023].

[48] «Determination of the homogeneity of an image,» [En ligne]. Available: https://www.medealab.de/en/2019/07/10/homogeneity-image-

processing/#:~:text=The%20distribution%20of%20gray%20and%2For%20color%20values%20within,strong%20contrasts%20within%20an%20image%2C%20it%20is%20inhomogeneous.. [Consulted on 3 6 2023].

[49] «Image Enhancement,» [En ligne]. Available: https://paperswithcode.com/task/image-enhancement. [Consulted on 3 6 2023].

[51] «What Is a Feature Descriptor in Image Processing?,» [En ligne]. Available: https://www.baeldung.com/cs/image-processing-feature-descriptors. [Consulted on 3 6 2023].

[52] «Image Feature Extraction: Traditional and Deep Learning Techniques,» [En ligne]. Available: https://towardsdatascience.com/image-feature-extraction-traditional-and-deep-learning-techniques-ccc059195d04. [Consulted on 3 6 2023].

[59] «Optimizing Performance: SelectKBest for Efficient Feature Selection in Machine Learning,» [En ligne]. Available: https://medium.com/@Kavya2099/optimizing-performance-selectkbest-for-efficient-feature-selection-in-machine-learning-3b635905ed48. [Consulted on 20 6 2023].

[61] «Introduction to Dimensionality Reduction,» [En ligne]. Available: https://www.geeksforgeeks.org/dimensionality-reduction/. [Consulted on 20 6 2023].

[63] «A Step-by-Step Explanation of Principal Component Analysis (PCA),» [En ligne]. Available: https://builtin.com/data-science/step-step-explanation-principal-component-analysis. [Consulted on 20 6 2023].

[66] «sklearn.random_projection.SparseRandomProjection,» [En ligne]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.random_projection.SparseRandomProjection.html#. [Consulted on 20 6 2023].

[68] «Artificial Intelligence. What Is Artificial Intelligence (AI)? How Does AI Work?,» [En ligne]. Available: https://builtin.com/artificial-intelligence. [Consulted on 13 6 2023].

[69] «artificial intelligence (AI),» [En ligne]. Available: https://www.techtarget.com/searchenterpriseai/definition/AI-Artificial-Intelligence. [Consulted on 13 6 2023].

[70] «What is Artificial Intelligence, Machine Learning, and Deep Learning?,» [En ligne]. Available: https://rapidminer.com/blog/artificial-intelligence-machine-learning-deep-learning/. [Consulted on 17 6 2023].

[71] «Artificial Intelligence: What It Is and How It Is Used,» [En ligne]. Available: https://www.investopedia.com/terms/a/artificial-intelligence-ai.asp#toc-types-of-artificial-intelligence. [Consulted on 13 6 2023].

[72] «What is machine learning?,» [En ligne]. Available: https://www.ibm.com/topics/machine-learning. [Consulted on 14 6 2023].

[73] «What Is Machine Learning? Definition, Types, and Examples.,» [En ligne]. Available: https://www.cqf.com/blog/what-machine-learning-definition-types-and-examples?gad=1&gclid=EAIaIQobChMIn6OCz67B_wIV1NbICh32OQypEAAYBCAAEgIR7fD_BwE. [Consulted on 14 6 2023].

[74] «Training data: the milestone of machine learning,» [En ligne]. Available: https://mapendo.co/blog/training-data-the-milestone-of-machine-learning. [Consulted on 17 6 2023].

[75] «What is Deep Learning? Simple Explained,» [En ligne]. Available: https://www.glweb.eu/blog/digital-transformation/201-what-is-deep-learning. [Consulted on 17 6 2023].

[76] «What are neural networks?,» [En ligne]. Available: https://www.ibm.com/topics/neural-networks. [Consulted on 14 6 2023].

[77] «API Update: ANN Bias Correction For Temperature Forecasts,» [En ligne]. Available: https://blog.weatherbit.io/api-new-bias-correction-algorithm/. [Consulted on 17 6 2023].

[78] «Machine learning, explained,» [En ligne]. Available: https://mitsloan.mit.edu/ideas-made-to-matter/machine-learning-explained. [Consulted on 14 6 2023].

[79] «The Ultimate Guide to Semi-Supervised Learning,» [En ligne]. Available: https://www.v7labs.com/blog/semi-supervised-learning-guide. [Consulted on 14 6 2023].

[80] «Self-Supervised Learning (SSL) Overview,» [En ligne]. Available: https://towardsdatascience.com/self-supervised-learning-ssl-overview-8a7f24740e40. [Consulted on 14 6 2023].

[81] «A Gentle Introduction to Transfer Learning for Deep Learning,» [En ligne]. Available: https://machinelearningmastery.com/transfer-learning-for-deep-learning/. [Consulted on 14 6 2023].

[82] «4 Types of Classification Tasks in Machine Learning,» [En ligne]. Available: https://machinelearningmastery.com/types-of-classification-in-machine-learning/. [Consulted on 24 6 2023].

[83] «Binary and Multiclass Classification in Machine Learning,» [En ligne]. Available: https://www.analyticssteps.com/blogs/binary-and-multiclass-classification-machine-learning. [Consulted on 24 6 2023].

[85] «KNN (K-Nearest Neighbors) #1,» [En ligne]. Available: https://towardsdatascience.com/knn-k-nearest-neighbors-1-a4707b24bd1d. [Consulted on 17 6 2023].

[86] «What is the k-nearest neighbors algorithm?,» [En ligne]. Available: https://www.ibm.com/topics/knn. [Consulted on 15 6 2023].

[87] «K-Nearest Neighbor(KNN) Algorithm,» [En ligne]. Available: https://www.geeksforgeeks.org/k-nearest-neighbours/. [Consulted on 15 6 2023].

[88] «Machine Learning Basics with the K-Nearest Neighbors Algorithm,» [En ligne]. Available: https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761. [Consulted on 15 6 2023].

[89] «Support Vector Machine Algorithm,» [En ligne]. Available: https://www.javatpoint.com/machine-learning-support-vector-machine-algorithm. [Consulted on 16 6 2023].

[91] [En ligne]. Available: https://github.com/ageron/handson-ml2/blob/master/05_support_vector_machines.ipynb. [Consulted on 17 6 2023].

[92] «What is Python? Executive Summary,» [En ligne]. Available: https://www.python.org/doc/essays/blurb/. [Consulted on 9 6 2023].

[93] «Anaconda,» [En ligne]. Available: https://www.dominodatalab.com/data-science-dictionary/anaconda. [Consulted on 9 6 2023].

[94] «Jupyter Notebook,» [En ligne]. Available: https://www.dominodatalab.com/data-science-dictionary/jupyter-notebook. [Consulted on 9 6 2023].

[95] «Pillow,» [En ligne]. Available: https://pillow.readthedocs.io/en/stable/. [Consulted on 9 6 2023].

[96] «Introduction,» [En ligne]. Available: https://docs.opencv.org/4.x/d1/dfb/intro.html. [Consulted on 9 6 2023].

[97] «OS Module in Python with Examples,» [En ligne]. Available: https://www.geeksforgeeks.org/os-module-python-examples/. [Consulted on 10 6 2023].

[98] «Matplotlib: Visualization with Python,» [En ligne]. Available: https://matplotlib.org/. [Consulted on 11 6 2023].

[99] [En ligne]. Available: https://scikit-learn.org/stable/. [Consulted on 12 6 2023].

[100] «Python time Module,» [En ligne]. Available: https://www.programiz.com/python-programming/time. [Consulted on 12 6 2023].