

DEEP LEARNING FOR BREAST CANCER PREDICTION

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Science in Bioinformatics and Data Science

By

B Viswa (41738017)



**DEPARTMENT OF BIOINFORMATICS
SCHOOL OF BIO AND CHEMICAL ENGINEERING**

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

**Accredited with Grade "A++" by NAAC | 12B Status by UGC | Approved by AICTE
Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai - 600 119**

APRIL – 2024



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Accredited with "A++" grade by NAAC I 12B Status by UGC I Approved by AICTE
Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai – 600 119

www.sathyabama.ac.in

DEPARTMENT OF BIOINFORMATICS

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **B Viswa (41738017)**, who carried out the final year project entitled "**DEEP LEARNING FOR BREAST CANCER PREDICTION**" under our supervision from January 2024 to April 2024.

DANIEL ALEX ANAND
Internal Guide

Dr. H. JEMMY CHRISTY
Head of the Department

Submitted for Viva Voce Examination held on 27/04/2024

Internal Examiner
External Examiner

DECLARATION

I, **B.Viswa**, hereby declare that the Project Report entitled **DEEP LEARNING FOR BREAST CANCER PREDICTION** done by me under the guidance of **Dr. ALEX ANAND DANIEL** at Sathyabama Institute of Science and Technology, Jeppiaar Nagar, Rajiv Gandhi Road, Chennai-600119, Tamil Nadu, India, is submitted in partial fulfillment of the requirements for the award of **Bachelor of Science in Bioinformatics and Data Science**.

DATE: 27/04/2024

PLACE: CHENNAI

பா. விஸ்வா

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENTS

I am delighted to express my heartfelt acknowledgement to the Board of Management of SATHYABAMA for their kind encouragement and support throughout the duration of my project. Their guidance has been invaluable in ensuring the successful completion of this project, and I am sincerely grateful for their contribution.

I convey my thanks to Dr. Rekha Chakravarthy, Dean, School of Science and Humanities for providing me necessary support and details at the right time during the progressive reviews.

I sincerely thank Dr. H. Jemmy Christy, Head of the Department, Dept. of Bioinformatics, for his unwavering support and timely provision of necessary details during the progressive reviews. His support has been instrumental in helping me to overcome obstacles and achieve success in this project.

I am deeply indebted to my Project Guide, Dr. Daniel Alex Anand, for his invaluable guidance, suggestions, and constant encouragement throughout the project. His expertise and insights have been instrumental in shaping my work and ensuring its successful completion.

I also wish to extend my gratitude to my family for their support and assistance in various ways, which have contributed to the successful completion of this project.

B.Viswa



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Accredited with "A++" grade by NAAC | 12B Status by UGC | Approved by AICTE
Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai – 600 119
www.sathyabama.ac.in

CERTIFICATE

is to certify that the dissertation entitled **"DEEP LEARNING FOR BREAST CANCER PREDICTION"** carried out by **Mr. B Viswa (Reg. No. 41738017)**, Department of Bioinformatics and Data Science, Sathyabama Institute of Science and Technology, Chennai in partial fulfillment of the requirements of the award of the **Degree of Bachelor of Science in Bioinformatics and Data Science 2021-2024**, is a true and faithful record of the project work done under my guidance and supervision. This thesis or part, therefore, has not been submitted elsewhere for any other degree or diploma.

Date: 27/04/2024

Place: CHENNAI

Dr. DANIEL ALEX ANAND

ABSTRACT

Breast cancer remains a significant health concern worldwide, emphasizing the need for advanced diagnostic tools to improve early detection and treatment outcomes. This thesis presents a novel deep learning-based approach for automated breast cancer detection using the Memorial Sloan Kettering Cancer Center (MSKCC) dataset in CSV format. The primary aim is to develop an accurate and efficient computational model capable of identifying malignant lesions from a comprehensive set of clinical and imaging features. The proposed methodology involves data preprocessing, feature extraction, and model development tailored to the characteristics of the MSKCC CSV dataset. Key clinical and imaging features, including patient demographics, tumor characteristics, and radiomic features extracted from mammographic images, are utilized as input to convolutional neural network (CNN) architectures. A systematic approach is adopted for data preprocessing, including missing data imputation, feature scaling, and normalization, to ensure optimal model performance. Deep learning models are trained using various CNN architectures, with hyperparameter optimization and cross-validation techniques employed to enhance model robustness and generalization. The performance of the developed models is extensively evaluated using metrics such as sensitivity, specificity, accuracy, and area under the receiver operating characteristic curve (AUC-ROC) on both training and independent validation datasets from the MSKCC CSV repository. Comparative analyses are conducted to assess the efficacy of the proposed deep learning-based approach compared to traditional diagnostic methods.

Keywords: *Breast Cancer (BC), Memorial Sloan Kettering Cancer Center (MSKCC), Deep Learning, CNN architecture , hyperparameter optimization , cross-validation technique.*

Table of Contents

LIST OF ABBREVIATIONS	vi
CHAPTER 1: INTRODUCTION	1
1.1 BACKGROUND	2
1.2 EXISTING PROBLEMS	3
1.3 TREATMENT	4
1.4 BOTTLENECK AREAS	6
CHAPTER 2: AIM AND SCOPE OF THE STUDY	7
3.1 AIM	7
3.2 SCOPE	8
CHAPTER 3: REVIEW OF LITERATURE	9
CHAPTER 4: METHODOLOGY	20
4.1 DATA SET	20
4.2 DATABASE	20
4.3 LOAD THE DATASET	21
4.4 TOOLS EMPLOYED	21
4.5 DATA PREDICTION	25
4.6 SPLIT THE DATA	25
4.7 INITIALIZE AND TRAIN	26
4.8 MODEL EVALUATION	27
4.9 PREDICTION	28
CHAPTER 5: RESULTS AND DISCUSSION	29
5.1 RESULTS	29
5.2 IMPORTING LIBRARIES	29
5.3 IMPORTING DATASET	30
5.4 UNDERSTANDING THE TYPE OF DATA	32
5.5 DATA CLEANING	37
5.6 DATA TRANSFORMATION	42
5.7 FEATURE ENGINEERING	43
5.8. DESCRIPTIVE STATISTICS	46
5.9 DATA VISUALIZATION	48

5.10 CHOOSING A MACHINE LEARNING ALGORITHM	52
5.11 MODEL TRAINING	53
5.12 MODEL EVALUATION	54
5.13 HYPERPARAMETER TUNING	55
5.14 MODEL COMPILATION	56
5.15 TENSOR SHAPE	57
5.16 BINARY CLASSIFICATION TASK	58
5.17 FLASK	60
CHAPTER 6: SUMMARY AND CONCLUSIONS-----	62
REFERENCES -----	64

LIST OF ABBREVIATIONS

CNN	Convolutional Neural Networks
MSKCC	Memorial Sloan Kettering Cancer Center
CSV	Comma Separated Values
HER2	Human Epidermal Growth Factor Receptor 2
TNBC	Triple Negative Breast Cancer
AUC	Area under the ROC Curve
ROC	Receiver Operating Characteristic
EGFR	Estimated Glomerular Filtration Rate
KRAS	Key Result Areas
MYC	Myelocytomatosis oncogene
CTCF	CCCTC binding factor
FOXA1	Forkhead Box A1
TBX3ER	T-box Transcription Factor 3
IDC	Invasive Ductal Carcinoma
ILC	Invasive Lobular Carcinoma
NF1	Neurofibromatosis type 1
ctDNA	Circulating Tumor DNA
TMB	Tumor Mutational Burden
TCGA	The Cancer Genome Atlas
ER	Estrogen Receptor
PR	Progesterone Receptor
TAM	Tumor-Associated Macrophages
AI	Aromatase Inhibitors
PFS	Progression Free Survival
DFS	Disease Free Survival
DRFS	Distant Recurrence-Free Survival
LBD	Ligand-Binding Domain
CDH1	E Cadherin
ML	Machine Learning
DL	Deep Learning

CAD	Computer Aided Diagnosis
FFDM	Full-Field Digital Mammographic
SVM	Support Vector Machine

LIST OF FIGURES

Figure 5.1	Importing Libraries
Figure 5.2	Importing Dataset
Figure 5.4	Importing Dataset
Figure 5.5	Number of Features
Figure 5.6	Types of Features
Figure 5.7	Categorical and Numerical
Figure 5.8	Feature Representation
Figure 5.9	Feature Representation Result
Figure 5.10	Check for missing values
Figure 5.11	Inconsistence data
Figure 5.12	Check for duplicate rows
Figure 5.13	Outliers
Figure 5.14	Normalize the data for Machine Learning Purpose
Figure 5.15	Create New Features
Figure 5.16	Modify Existing ones
Figure 5.17	Improve Model Performance

Figure 5.18	Generating Mean
Figure 5.19	Target the feel of the data
Figure 5.20	Line Plot
Figure 5.21	Box Plot
Figure 5.22	Scatter Plot
Figure 5.23	Correlation Heat Map Analysis
Figure 5.24	Choosing a Machine Learning Algorithm Code
Figure 5.25	Choosing a Machine Learning Algorithm Output
Figure 5.26	Model Training
Figure 5.27	Model Evaluation
Figure 5.28	Hyperparameter Tuning
Figure 5.29	Model Compilation
Figure 5.30	Tensor Shape
Figure 5.31	Binary Classification Task
Figure 5.32	Flask

CHAPTER 1

INTRODUCTION

Breast cancer is a significant global health challenge, accounting for a substantial portion of cancer-related morbidity and mortality among women worldwide. Early detection plays a pivotal role in improving patient outcomes by enabling timely intervention and treatment. Mammography, the gold standard for breast cancer screening, has limitations in terms of interpretation and diagnostic accuracy, often leading to missed or delayed diagnoses.

Breast cancer arises from the uncontrolled growth of abnormal cells in the breast tissue, which can manifest as a lump or mass that may be detected through screening or self-examination. While the majority of breast cancer cases occur in women, it is important to acknowledge that men can also develop this disease, albeit at a significantly lower frequency.

In recent years, advances in deep learning techniques have shown promise in improving the accuracy and efficiency of breast cancer detection from mammographic images. By leveraging convolutional neural networks (CNNs) and large-scale datasets, deep learning models can automatically extract relevant features and identify suspicious lesions indicative of breast malignancy.

The Memorial Sloan Kettering Cancer Center (MSKCC) dataset, available in CSV format, offers a valuable resource for research in breast cancer detection. Comprising a comprehensive collection of clinical and imaging data, including patient demographics, tumor characteristics, and radiomic features extracted from mammographic images, this dataset provides a rich source of information for training and evaluating deep learning models.

1.1 BACKGROUND

Breast cancer is a heterogeneous disease characterized by diverse molecular subtypes, clinical behaviors, and treatment responses. Despite advancements in diagnostic and therapeutic strategies, breast cancer remains a leading cause of cancer-related mortality among women globally. The complexity of breast cancer arises from its multifactorial etiology, encompassing genetic predisposition, hormonal influences, lifestyle factors, and environmental exposures. Understanding the underlying biology and molecular mechanisms driving breast cancer development and progression is essential for improving patient outcomes and developing targeted therapies.

In recent years, there has been a growing interest in leveraging computational methods, particularly deep learning algorithms, for the analysis of medical imaging data in breast cancer diagnosis and prognosis. Mammography, the primary imaging modality for breast cancer screening, produces high-resolution images that are traditionally interpreted by radiologists. However, the subjective nature of interpretation, coupled with inter-observer variability and the potential for human error, can impact the accuracy and consistency of breast cancer diagnosis.

Deep learning, a subset of artificial intelligence, offers a promising solution to address these challenges by automating the analysis of mammographic images and extracting relevant features indicative of breast malignancy. Convolutional neural networks (CNNs), a type of deep learning architecture, have demonstrated remarkable success in image recognition tasks and have been increasingly applied to medical imaging datasets, including mammography.

The Memorial Sloan Kettering Cancer Center (MSKCC) dataset provides a valuable resource for researchers in the field of breast cancer detection and diagnosis. Comprising a diverse collection of mammographic images along with detailed clinical and pathological annotations, the MSKCC dataset enables the development and evaluation of deep learning models for automated breast cancer detection.

1.2 EXISTING PROBLEMS

Late Diagnosis: One of the most significant challenges in breast cancer management is late diagnosis. Many cases are not detected until the disease has progressed to advanced stages, when treatment options may be limited and prognosis is poorer. Improving early detection methods and increasing awareness about the importance of screening are crucial in addressing this issue.

Treatment Resistance: While various treatment modalities, such as surgery, chemotherapy, radiation therapy, hormonal therapy, targeted therapy, and immunotherapy, have improved survival rates, resistance to these treatments remains a significant problem. Tumors often develop resistance to chemotherapy or targeted therapies, leading to treatment failure and disease recurrence. Understanding the mechanisms of resistance and developing strategies to overcome it are essential for improving treatment outcomes.

Lack of Targeted Therapies for Certain Subtypes: While targeted therapies have revolutionized the treatment of certain breast cancer subtypes, such as HER2-positive or hormone receptor-positive breast cancer, there is still a need for effective targeted therapies for other subtypes, such as triple-negative breast cancer (TNBC). TNBC tends to be more aggressive and has limited treatment options compared to other subtypes, highlighting the need for further research into targeted therapies for this subtype.

Long-term Survivorship Challenges: As survival rates for breast cancer improve, there is a growing population of breast cancer survivors facing long-term survivorship challenges, such as the late effects of treatment, risk of recurrence, and ongoing physical and emotional symptoms. Developing survivorship care plans and providing comprehensive follow-up care are essential for addressing the long-term needs of breast cancer survivors.

Lack of Funding and Collaboration: Despite the progress made in breast cancer research, there is still a need for sustained funding and collaboration to support innovative

research initiatives, translate scientific discoveries into clinical practice, and improve patient outcomes. Collaboration between researchers, clinicians, advocacy organizations, policymakers, and industry stakeholders is essential for advancing breast cancer research and improving patient care.

Limited options for high-risk patients: Children with high-risk AML may not respond well to standard treatments and may have limited options for effective therapy.

Lack of personalized medicine: There is a need for more personalized medicine approaches to treatment, as the genetic makeup of each patient's AML may play a role in treatment success.

1.3 TREATMENT

Treating Breast Cancer (BC) typically involves a combination of surgery, radiation therapy, chemotherapy, hormonal therapy, targeted therapy, and immuno therapy. The specific treatment plan will depend on the individual patient .

Surgery: Surgery is often the primary treatment for breast cancer and involves removing the tumor and surrounding tissue. Types of surgical procedures used in breast cancer treatment include lumpectomy (breast-conserving surgery), mastectomy (removal of the entire breast), and lymph node dissection (removal of nearby lymph nodes).

Radiation Therapy: Radiation therapy uses high-energy beams to destroy cancer cells and shrink tumors. It is commonly used after surgery to kill any remaining cancer cells and reduce the risk of recurrence. Radiation therapy may also be used before surgery to shrink tumors or as a palliative treatment to relieve symptoms in advanced cases.

Chemotherapy: Chemotherapy involves the use of drugs to kill cancer cells or stop them from growing and dividing. It is often used in combination with surgery and/or radiation therapy to treat early-stage or locally advanced breast cancer. Chemotherapy may also be used as the primary treatment for metastatic breast cancer (cancer that has spread to other parts of the body).

Hormonal Therapy: Hormonal therapy, also known as endocrine therapy, targets hormone receptors on cancer cells to block the effects of estrogen and/or progesterone, hormones that can fuel the growth of certain types of breast cancer. Hormonal therapy is commonly used in hormone receptor-positive breast cancer and may be used as adjuvant therapy after surgery or as the primary treatment for metastatic breast cancer.

Targeted Therapy: Targeted therapy specifically targets cancer cells by interfering with specific molecules involved in cancer growth and progression. Examples of targeted therapies used in breast cancer treatment include HER2-targeted therapies (such as trastuzumab and pertuzumab) for HER2-positive breast cancer and CDK4/6 inhibitors (such as palbociclib and ribociclib) for hormone receptor-positive, HER2-negative breast cancer.

Immunotherapy: Immunotherapy harnesses the body's immune system to recognize and destroy cancer cells. While not yet as widely used as other treatments, immunotherapy has shown promise in certain types of breast cancer, particularly triple-negative breast cancer (TNBC). Immune checkpoint inhibitors, such as pembrolizumab and atezolizumab, are being studied in clinical trials for the treatment of breast cancer.

Clinical trials: Some breast cancer patients may be eligible for participation in clinical trials of new and innovative treatments.

1.4 BOTTLENECK AREAS

Treatment Resistance: Despite the effectiveness of various treatment modalities, many patients with breast cancer develop resistance to therapy over time, leading to disease progression and poorer outcomes. Understanding the mechanisms of treatment resistance and developing strategies to overcome it are critical for improving treatment efficacy and patient survival.

Side Effects and Toxicities: Breast cancer treatments, such as chemotherapy, radiation therapy, and targeted therapy, often cause significant side effects and toxicities that can impact patients' quality of life. Common side effects include fatigue, nausea, hair loss, neuropathy, and cognitive dysfunction. Managing these side effects and developing interventions to minimize treatment-related toxicities are important considerations in breast cancer care.

Healthcare Disparities: Disparities in access to breast cancer screening, diagnosis, and treatment exist based on factors such as race, ethnicity, socioeconomic status, geographic location, and healthcare infrastructure. Addressing healthcare disparities and ensuring equitable access to high-quality care and treatment options are essential for reducing disparities in breast cancer outcomes.

CHAPTER 2

AIM AND SCOPE OF THE STUDY

3.1 AIM

This project aims to develop and evaluate a deep learning-based approach for automated breast cancer detection from mammographic images. Using the Memorial Sloan Kettering Cancer Center (MSKCC) dataset in CSV format, the project seeks to design convolutional neural network (CNN) architectures tailored for breast cancer detection tasks. Emphasis will be placed on exploring data preprocessing techniques to enhance dataset quality, including normalization, augmentation, and noise reduction. Additionally, transfer learning strategies will be investigated to leverage pre-trained CNN models for improved performance and generalization.

The project aims to train and optimize deep learning models using the MSKCC CSV dataset, with a focus on achieving high sensitivity, specificity, and area under the receiver operating characteristic curve (AUC-ROC) on independent validation datasets. Evaluation will involve rigorous validation metrics and comparative analyses against existing methods to assess accuracy, efficiency, and clinical relevance.

Further, the project will discuss the clinical implications and potential challenges associated with deploying deep learning-based breast cancer detection systems in real-world healthcare settings, addressing interpretability, scalability, and regulatory considerations.

This project aims to contribute to advancing computer-aided diagnosis in breast cancer and improving early detection rates and patient outcomes in the management of this disease.

3.2 SCOPE

This project focuses on the development and evaluation of a deep learning-based approach for automated breast cancer detection from mammographic images using the Memorial Sloan Kettering Cancer Center (MSKCC) dataset in CSV format.

The scope of the project includes the design and implementation of convolutional neural network (CNN) architectures tailored specifically for breast cancer detection tasks. Data preprocessing techniques such as normalization, augmentation, and noise reduction will be explored to enhance the quality and consistency of the MSKCC CSV dataset. Additionally, the project will investigate transfer learning strategies to leverage pre-trained CNN models and adapt them to the task of breast cancer detection, aiming to improve model performance and generalization.

The project will involve training and optimizing deep learning models using the MSKCC CSV dataset, with a focus on achieving high sensitivity, specificity, and area under the receiver operating characteristic curve (AUC-ROC) on independent validation datasets. Evaluation metrics will include rigorous validation methods and comparative analyses against existing methods to assess accuracy, efficiency, and clinical relevance.

However, the project does not encompass clinical trials or direct patient care, and the deployment of deep learning-based breast cancer detection systems in real-world healthcare settings will be discussed primarily from a conceptual and regulatory perspective. The scope of the project is limited to the development and evaluation of computational models for breast cancer detection and does not extend to treatment decision-making or outcome prediction.

CHAPTER 3

REVIEW OF LITERATURE

The primary purpose of a literature review is to provide a comprehensive understanding of the existing knowledge on a particular topic or subject. It helps to identify gaps in the current knowledge, highlight areas for further research, and establish a theoretical framework for the study.

A literature review typically involves a thorough search of relevant literature and research databases, followed by a careful analysis of the findings. The review may be conducted as part of a larger research project, or as a standalone piece of work, such as a dissertation or thesis.

The following are the various references reviewed for the purpose of the current study.

- **The Genomic Landscape of Endocrine-Resistant Advanced Breast Cancers-
Pedram Razavi et al.,2018**

The study revealed an increased frequency of genetic alterations in genes related to the MAPK pathway and the transcriptional machinery of the estrogen receptor in tumors that were resistant to hormonal therapy. Specifically, they found that mutations activating the ERBB2 gene and loss-of-function mutations in the NF1 gene were more prevalent in these resistant tumors compared to non-resistant ones. Additionally, mutations in other genes involved in the MAPK pathway (such as EGFR and KRAS) and estrogen receptor transcriptional regulators (including MYC, CTCF, FOXA1, and TBX3) were also more common in resistant tumors.

These alterations were observed in approximately 22% of tumors and were found to be mutually exclusive with mutations in the ESR1 gene. Further, they were associated with a shorter duration of response to subsequent hormonal therapies.

This study provides insights into the genetic factors contributing to resistance to hormonal therapy in breast cancer, highlighting the potential importance of the MAPK pathway and estrogen receptor transcriptional machinery in this process.

- **Loss of function of NF1 is a mechanism of acquired resistance to endocrine therapy in lobular breast cancer - E S Sokol et.,2019**

A study focused on understanding the genomic landscape of invasive lobular carcinoma (ILC), particularly in the context of resistance to endocrine therapy. While previous studies have primarily examined early-stage ILC, this research focuses on refractory cases to provide insights into acquired resistance mechanisms.

The study analyzed tissue samples from 336 primary ILC cases, 485 estrogen receptor (ER)-positive invasive ductal carcinoma (IDC) cases, and metastatic biopsy specimens from 180 ILC and 191 ER-positive IDC patients. Using comprehensive genomic profiling, alterations in up to 395 cancer-related genes were assessed, including short variants, indels, copy number variants, and rearrangements.

The findings indicate that while ESR1 alterations are enriched in metastatic ILC and IDC, NF1 alterations are specifically enriched in metastatic ILC (mILC). These NF1 alterations often occur under loss of heterozygosity and are mutually exclusive with ESR1 mutations. They are also frequently polyclonal in circulating tumor DNA (ctDNA) assays, suggesting their emergence in acquired resistance scenarios. Moreover, in vitro experiments demonstrated that NF1 knockdown confers a growth advantage in the presence of tamoxifen.

The study also observed a significant increase in tumor mutational burden (TMB) in mILCs compared to primary ILCs or metastatic IDCs, with most TMB-high mILCs exhibiting an APOBEC trinucleotide signature.

Overall, the study identifies NF1 alteration as a potential driver of endocrine therapy resistance in mILC, suggesting the potential benefit of RAS/RAF kinase inhibitors. Additionally, the elevated TMB in a subset of mILCs suggests a possible benefit from immune checkpoint inhibitors.

This research provides valuable insights into the genomic alterations associated with endocrine therapy resistance in ILC, paving the way for targeted therapeutic strategies and potential immunotherapeutic interventions.

• Genomic Landscape and Endocrine-Resistant Subgroup in Estrogen Receptor-Positive, Progesterone Receptor-Negative, and HER2-Negative Breast Cancer

- Xi-Yu Liu et al.,2018

The study is focused on understanding the multiomic landscape and endocrine responsiveness of a specific subtype of breast cancer known as estrogen receptor-positive, progesterone receptor-negative, and human epidermal growth factor receptor 2-negative (ER+PR-HER2-) breast cancer. This subtype constitutes approximately 10% of all breast cancer cases and is known to benefit less from endocrine therapy compared to other subtypes.

The study utilized five cohorts, including large databases such as the Surveillance, Epidemiology, and End Results (SEER) database, the Molecular Taxonomy of Breast

Cancer International Consortium, The Cancer Genome Atlas (TCGA), and datasets from MD Anderson and Fudan University Shanghai Cancer Center.

Clinically, ER+PR-HER2- tumors exhibited lower responsiveness to endocrine therapy compared to ER+PR+HER2- tumors. Genomically, the study found that a significant proportion of ER+PR-HER2- tumors had copy number loss or promoter methylation of PR genes, which explained the loss of progesterone receptor expression. These tumors also showed higher rates of TP53 mutations and lower rates of PIK3CA mutations compared to ER+PR+HER2- tumors. Additionally, amplification events in genes such as ZNF703 and RPS6KB1 were more common in ER+PR-HER2- tumors.

A subset of ER+PR-HER2- tumors (~20%) were classified as non-luminal-like based on gene expression analysis, exhibiting lower endocrine sensitivity scores and enrichment in biosynthesis, metabolism, and DNA replication pathways. This non-luminal-like subgroup was further identified using immunohistochemistry markers (GATA3, CK5, and EGFR) and was found to derive limited benefit from adjuvant endocrine therapy.

ER+PR-HER2- breast cancer comprises clinically and genomically distinct groups that may necessitate different treatment approaches. The identification of a non-luminal-like subgroup with reduced responsiveness to endocrine therapy underscores the importance of personalized treatment strategies for this subtype of breast cancer.

- **ESR1 mutations are frequent in newly diagnosed metastatic and loco-regional recurrence of endocrine-treated breast cancer and carry worse prognosis**
- Adi Zundeleovich et al.,2020

It presents a study which is focused on understanding the incidence and clinical implications of ESR1 mutations, associated with resistance to endocrine therapy, in early recurrence events of breast cancer, such as local recurrences or newly diagnosed metastatic disease.

The study collected tumor samples from 103 breast cancer patients who had received endocrine therapy before experiencing local or metastatic recurrence. The cohort included patients with local/loco-regional recurrence and those with metastatic disease, both newly diagnosed and advanced. The prevalence of ESR1 hotspot mutations, including D538G, L536R, and Y537S/N/C, was analyzed using targeted sequencing or droplet digital PCR.

The results showed that ESR1 mutations were present in 12% of newly diagnosed metastatic patients and 18% of advanced metastases, detected at allele frequency > 1%. These mutations were predominantly found in patients previously treated with both tamoxifen (TAM) and aromatase inhibitors (AI). Additionally, in the local recurrence cohort, ESR1 mutations were identified in 36% of patients, with 10% detected at allele frequency > 1%. Notably, one patient developed an ESR1 mutation while on neoadjuvant endocrine therapy.

Patients with ESR1 mutations experienced significantly shorter progression-free survival (PFS) on AI treatment in the metastatic setting, and disease-free survival (DFS) and distant recurrence-free survival (DRFS) were also significantly shorter in patients with ESR1 mutations in their loco-regional recurrence tumors.

It is a clinically relevant ESR1 mutations are prevalent in both newly diagnosed metastatic and local recurrence of endocrine-treated breast cancer. Given that local recurrences are

amenable to curative therapy, these mutations may guide the selection of subsequent endocrine therapies.

• **Recurrent hyperactive ESR1 fusion proteins in endocrine therapy-resistant breast cancer - R J Hartmaier et al.,2018**

It sheds light on different aspects of endocrine therapy resistance in estrogen receptor-positive (ER-positive) metastatic breast cancer, highlighting the significance of genomic alterations in ESR1.

It focuses on the identification of recurrent ESR1 fusion proteins in advanced breast cancer, elucidating their role in endocrine therapy resistance. Through multimodal sequencing efforts across three patient cohorts, the study identifies 88 ESR1 rearrangements (REs) in ER-positive metastatic disease. Importantly, the fusions lack an intact ligand binding domain (LBD), rendering them impervious to ER-targeted therapies. This finding suggests that N-terminal ESR1 fusions involving exons 6-7 are a recurrent driver of endocrine therapy resistance, contributing valuable insights into potential treatment strategies.

It investigates the prevalence and clinical implications of ESR1 mutations in early recurrence events, such as local recurrences or newly diagnosed metastatic disease. The study collects tumor samples from breast cancer patients treated with endocrine therapy prior to local or metastatic recurrence. It identifies clinically relevant ESR1 mutations in both newly diagnosed metastatic and locally recurrent breast cancer, indicating their potential role in driving resistance to endocrine therapy. These mutations, particularly when present at high allele frequencies, are associated with significantly shorter progression-free survival, disease-free survival, and distant recurrence-free survival.

So It underscores the importance of understanding the genomic landscape of ER-positive breast cancer and its implications for treatment resistance. They highlight the significance of ESR1 alterations, whether through fusion proteins or mutations, in driving

endocrine therapy resistance, and emphasize the need for personalized therapeutic approaches informed by genomic profiling.

• The association between type of endocrine therapy and development of estrogen receptor-1 mutation(s) in patients with hormone-sensitive advanced breast cancer: A systematic review and meta-analysis of randomized and non-randomized trials - Omar Najim et al.,2019

The aim is to assess the impact of Aromatase Inhibitors (AIs), Tamoxifen (TAM), and Fulvestrant (FUL) on the development of ESR1 mutations in hormone-sensitive advanced breast cancer. Through a systematic review of qualitative studies published between 2007 and 2019, the review identifies 16 relevant articles, including randomized controlled trials (RCTs) and cohort studies, encompassing a total of 2632 patients.

It reveals that ESR1 mutations are present in approximately 24% of patients with hormone-sensitive advanced breast cancer. Among these mutations, D538G is the most frequent. Importantly, the review highlights that prior exposure to endocrine therapy, particularly AIs, TAM, and FUL, can lead to the development of ESR1 mutations and subsequent therapy resistance, resulting in disease progression or recurrence.

Several mechanisms underlying endocrine therapy resistance are discussed, with a notable finding being the significantly higher incidence of ESR1 mutations in patients who received AI-containing therapeutic regimens compared to those who did not. This suggests that AI therapy may contribute more strongly to the development of ESR1 mutations and subsequent resistance.

The point is to underscore the importance of understanding the impact of different endocrine therapies on the development of ESR1 mutations in hormone-sensitive advanced breast cancer. It highlights the need for tailored treatment approaches considering the risk of ESR1 mutation-associated resistance, particularly in patients receiving AI therapy.

- **Frequent somatic CDH1 loss-of-function mutations in plasmacytoid variant bladder cancer**

This highlights findings from a study investigating plasmacytoid bladder cancer, a particularly aggressive histological variant associated with high disease-specific mortality. Through whole-exome and targeted sequencing, the study identifies truncating somatic alterations in the CDH1 gene in 84% of plasmacytoid carcinomas, indicating a high prevalence specific to this histological subtype.

Given the aggressive clinical behavior of plasmacytoid carcinomas, which often recur locally, the study further explores the functional implications of CDH1 alterations. Using CRISPR/Cas9-mediated knockout of CDH1 in bladder cancer cells, the researchers observe enhanced cell migration, suggesting a potential mechanistic link between CDH1 alterations and the aggressive behavior of plasmacytoid bladder cancer.

This study provides valuable insights into the genetic landscape of plasmacytoid bladder cancer and its potential impact on disease progression and aggressiveness, laying the groundwork for further research into targeted therapeutic strategies for this challenging histological variant.

Al-Ahmadie HA, Iyer G, Lee BH, Scott SN, Mehra R, Bagrodia A, Jordan EJ, Gao SP, Ramirez R, Cha EK, et al. (2016). Frequent somatic CDH1 loss-of-function mutations in plasmacytoid variant bladder cancer. Nat Genet 48, 356–358. - [PMC](#) - [PubMed](#)

- **Survey on Machine Learning and Deep Learning Applications in Breast Cancer Diagnosis.**

This survey is focused on the application of artificial intelligence (AI) technologies, particularly machine learning (ML) and deep learning (DL), in diagnosing breast carcinoma. Given that breast cancer is a leading cause of cancer-related deaths in females, early diagnosis and prognosis are crucial for improving survival rates. The survey reviews newly developed computer-aided diagnosis (CAD) systems that utilize ML and DL techniques, comparing them with established methods and discussing their technical details, advantages, and disadvantages.

The survey also addresses open issues, research gaps, and future directions in advanced CAD models for medical image analysis. Over the past decade, ML and DL have emerged as powerful tools in AI, offering promising results in healthcare applications with improved efficiency and reduced costs. By analyzing various classifiers within ML and DL approaches for breast cancer diagnosis, the survey highlights the superior performance of DL over conventional ML methods, particularly when dealing with large datasets.

The survey emphasizes the urgent need for practical and scientific research to address existing gaps and enhance healthcare outcomes in the long term. Overall, it provides a comprehensive overview of the state-of-the-art AI-based approaches for breast cancer diagnosis, aiming to contribute to ongoing advancements in the field and ultimately improve patient outcomes..

Gunjan Chugh., Shailender Kumar., Nanhay Singh. (2021). Survey on Machine Learning and Deep Learning Applications in Breast Cancer Diagnosis, 13, 1451–1470.

- **Breast cancer detection using deep learning: Datasets, methods, and challenges ahead.**

A comprehensive research endeavor focused on the early detection and classification of breast cancer (BC) using various imaging modalities, including mammography, histopathology, ultrasound, PET/CT, MRI, and thermography. Despite advancements in radiographic and histopathological imaging, the analysis by experienced professionals remains costly and error-prone.

The transformative potential of computer vision and machine learning (ML), particularly deep learning (DL), in biomedical applications due to its ability to efficiently handle large and complex datasets. DL techniques have shown promising results in identifying and prognosing BC using radiographic and histopathological images. However, most findings are limited to retrospective studies, necessitating external validation for clinical application.

The main objective of this research is to critically analyze existing research on BC detection and classification using ML, DL, and deep reinforcement learning. The review encompasses past research papers and publicly available datasets across various imaging modalities. Additionally, the abstract includes a discussion on open research challenges and prospects for future studies in this emerging field, highlighting the limitations of DL approaches.

This research aims to contribute to the ongoing advancements in BC diagnosis and classification by providing a critical analysis of existing methodologies and suggesting future research directions.

Nusrat Mohi ud din., Rayees Ahmad Dar., Muzafar Rasool., Assif Assad (2022). Breast cancer detection using deep learning: Datasets, methods, and challenges ahead.

<https://doi.org/10.1016/j.compbiomed.2022.106073>

- **Deep learning in breast cancer risk assessment: evaluation of convolutional neural networks on a clinical dataset of full-field digital mammograms**

A study evaluating the efficacy of deep learning, specifically convolutional neural networks (CNNs) with transfer learning, in assessing breast cancer risk directly from full-field digital mammographic (FFDM) images. The study includes 456 clinical FFDM cases categorized into three datasets: BRCA1/2 gene-mutation carriers (high-risk), unilateral cancer patients (high-risk), and low-risk individuals.

The performance of deep learning is compared to that of computerized radiographic texture analysis (RTA) and a combination of both methods in distinguishing between high- and low-risk subjects. The results indicate that CNNs perform similarly to RTA in distinguishing BRCA1/2 carriers and low-risk women, but outperform RTA in distinguishing unilateral cancer patients and low-risk women. Moreover, fusion classifiers combining deep learning and RTA features achieve significantly better performance than RTA alone in both scenarios.

The study concludes that deep learning, by extracting parenchymal characteristics from FFDMs, performs as well as or better than conventional texture analysis in differentiating between different cancer risk populations. This suggests the potential of deep learning techniques in improving breast cancer risk assessment using mammographic images.

Hui Li, Maryellen L. Giger, Benjamin Q. Huynh, Natalia O. Antropova et al.(2017). Deep learning in breast cancer risk assessment: evaluation of convolutional neural networks on a clinical dataset of full-field digital mammograms. PMCID: PMC5596198 PMID: 28924576

CHAPTER 4

METHODOLOGY

4.1 DATA SET

Gene data sets were downloaded from cbiportal database, and the vast data was filtered according to our parameters using Microsoft Excel.

4.2 DATABASE

cBioPortal- A Web Platform for Gene-Based Exploration (<https://www.cbioportal.org/>)

This is a database online tool which does analysis based on the data present in the database to the personalised input data. *The cBio Cancer Genomics Portal (<http://cbioportal.org>) is an open-access resource for the interactive exploration of multidimensional cancer genomics data sets, currently providing access to data from more than 5,000 tumour samples from 20 cancer studies. The cBio Cancer Genomics Portal significantly lowers the barriers between complex genomic data and cancer researchers who want rapid, intuitive, and high-quality access to molecular profiles and clinical attributes from large-scale cancer genomics projects and empowers researchers to translate these rich data sets into biologic insights and clinical applications. Cancer Discov; 2(5); 401–4. ©2012 AACR. (Cerami et al., 2012 & Gao et al., 2013)*

It gave insights into the mutation frequency, the structural variation, CN segments etc.

4.3 LOAD THE DATASET

It is important to use reliable sources such as the cbiportal or other reputable databases when downloading databases related to diabetes, heart disease, and breast cancer. Here is a section that describes the process of downloading dataset .

4.3.1 BREAST CANCER

For breast cancer prognosis, the cBioPortal Breast Cancer datasets provide valuable data to load into Python. environments Using pandas or similar data loading tools, researchers can import a dataset. Loading data into a DataFrame allows researchers to preprocess data, handle missing values, and code categorical variables before training machine learning models to predict breast cancer.

In all cases, loading data into a DataFrame allows for efficient operation using Python libraries. data processing, research and preparation of machine learning tasks. Researchers should ensure data integrity, verify feature significance and follow best practices.

4.4 TOOLS EMPLOYED

4.4.1 PYTHON

Python (<https://www.python.org/>) is a versatile programming language that has become indispensable in the fields of biology and bioinformatics due to its simplicity and powerful suite of libraries. In biological research, Python facilitates complex data analysis, visualisation, and simulation tasks, making it an essential tool for processing large datasets, such as genomic sequences and protein structures. Libraries like Pandas and NumPy streamline data manipulation and statistical analysis, while Matplotlib and Seaborn enable sophisticated data visualisation, critical for interpreting biological data. 4.1.3 PANDAS Pandas is a sophisticated Python library with strong data manipulation and analysis capabilities, and it is a mainstay in biological and bioinformatics research. Because of its smooth handling of tabular data, it is a vital tool for high-throughput

experiments and genomic studies for tasks like quality control, normalisation, and data integration. Pandas is an essential bioinformatics tool for processing and analysing data related to transcriptomics, proteomics, and genomics. It helps researchers find patterns and derive valuable insights from large, complicated biological datasets.

4.4.2 SEABORN

Seaborn is a Python visualisation toolkit that is highly integrated with pandas data structures and built upon Matplotlib, making it a powerful tool for bioinformatics and biological data processing. Researchers can effectively explore and understand patterns, trends, and relationships in genetic, proteomic, and other biological datasets with Seaborn's features, which support distribution analysis, advanced plotting functions for categorical variables, and the plotting of high-dimensional data. This skill is essential in the field of bioinformatics, as it facilitates the formulation of hypotheses, data-driven decision-making, and the dissemination of scientific discoveries.

4.4.3 MATPLOTLIB

Matplotlib is a flexible Python charting package that is essential for the visualisation of large and complicated data sets in the fields of biology and bioinformatics. This is an essential tool for anyone using Python to create a broad variety of static, interactive, and animated visualisations. Matplotlib is a tool used in bioinformatics and biological research that makes it easier to analyse and display massive biological data sets, including genetic association data, protein structures, and genomic sequences. Plotting complex trends and correlations in data is a valuable tool for biologists and bioinformaticians to find underlying patterns, compare diverse data sets, and effectively convey findings.

4.4.4 PYLPOT

Pyplot is a module of Matplotlib, a complete library for creating static, animated and interactive visualisations in Python. In the field of biology and bioinformatics, Pyplot plays a key role because it allows researchers to plot complex datasets with high accuracy and flexibility. It facilitates the visualisation of biological data such as gene expression profiles, metabolic pathways or evolutionary relationships, making it easier to identify patterns, trends and anomalies. Pyplot's ability to produce a wide variety of graphs - from simple line plots to complex heatmaps - makes it an essential tool for exploring, presenting and interpreting data in biological research and genomics, improving understanding of biological processes and systems at the molecular level.

4.4.5 NUMPY

NumPy, or Numerical Python, stands as a cornerstone in the toolkit of scientific computing within Python. Its primary offering is a powerful array object, facilitating the manipulation of homogeneous multidimensional data effortlessly. Through a rich suite of mathematical functions and operators, NumPy empowers efficient numerical computations, including arithmetic operations, statistical analyses, and array transformations. One of its standout features is broadcasting, which allows for seamless integration of arrays with different shapes, streamlining vectorized operations.

4.4.5 SCIPY

SciPy, a companion library to NumPy, is an indispensable tool in scientific computing and technical research within Python. Building upon NumPy's foundation, SciPy offers a vast array of high-level functions and algorithms for numerical optimization, integration, interpolation, and much more. Its extensive collection of submodules covers a wide range of scientific disciplines, including signal processing, statistics, and linear algebra. SciPy's

integration with NumPy facilitates seamless data exchange and manipulation, making it an integral part of the Python scientific ecosystem.

4.4.5 STATSMODELS

Statsmodels is a powerful Python library for estimating and interpreting statistical models. It offers a comprehensive range of functionalities for statistical analysis, including regression analysis, time-series analysis, hypothesis testing, and more. With its intuitive API and extensive documentation, Statsmodels facilitates the exploration and understanding of complex datasets, enabling researchers to derive meaningful insights from their data. Its integration with Pandas for data manipulation and Matplotlib for visualization enhances its utility in exploratory data analysis and model diagnostics.

4.4.6 SCIKIT-LEARN

Scikit-learn, often abbreviated as sklearn, stands as a premier machine learning library in Python, renowned for its simplicity, efficiency, and versatility. Offering a comprehensive suite of algorithms for classification, regression, clustering, dimensionality reduction, and more, sklearn empowers researchers and practitioners to build and deploy machine learning models with ease. Its consistent API and extensive documentation make it accessible to users of all skill levels, facilitating rapid prototyping and experimentation. Sklearn also provides tools for model evaluation, hyperparameter tuning, and model selection, enabling rigorous assessment and optimization of predictive models. Leveraging sklearn's integration with other Python libraries like NumPy and Pandas, researchers can seamlessly integrate machine learning workflows into their data analysis pipelines.

4.5 DATA PREPARATION

Data preparation is a critical step in building machine learning models to predict diseases related to diabetes, heart disease and breast cancer. This section describes the data preparation process for each disease.

4.5.1 BREAST CANCER

For breast cancer prognosis, data preparation focuses on data such as mammography findings, age, family history, hormonal factors, and tumor characteristics of a dataset. such as the cBioPortal Breast Cancer Database. Data is cleaned by dealing with missing values and outliers, and feature engineering may involve creating derived functions or scaling numerical variables for consistency. Categorical variables such as tumor type or hormone receptor status are coded to ensure that they contribute significantly to predictive models.

In general, data preparation for breast cancer prediction tasks involves dealing with data quality issues and missing values. and outliers, coding categorical variables, and dataset preparation for machine learning models. A thorough data preparation process improves the predictive power and generalizability of machine learning models, leading to more accurate and reliable disease prediction results.

4.6 SPLIT THE DATA

Dividing the data into training and test sets is a crucial step in developing a machine learning model to predict breast cancer. Here is a section that describes the data splitting process.

4.6.1 BREAST CANCER

For breast cancer prediction, datasets such as the cBioPortal Breast Cancer dataset are shared. to training and test series following the same principles. The training set is used to train machine learning models using features such as mammography findings, age, family history, hormonal factors and tumor characteristics. The test set is then used to evaluate the performance of the trained models in predicting breast cancer outcomes (such as malignancy or tumor type) based on unseen patient data.

By splitting the data into training and test sets, machine learning practitioners can. effectively develop and evaluate predictive models, ensuring that the models generalize well to new data and perform accurately in real cases. In disease prediction scenarios.

4.7 INITIALIZE AND TRAIN

Running and training machine learning models is an integral part of disease prediction tasks for diabetes, heart disease and breast cancer. This chapter presents the initial and training models for each disease.

4.7.1 BREAST CANCER

Machine learning models such as logistic regression for breast cancer prediction. , support vector machines (SVMs) or neural networks are initialized and trained using the cBioPortal breast cancer dataset. Features such as mammographic findings, age, family history, hormonal factors and tumor characteristics are used to train models to predict breast cancer outcomes, such as malignancy or tumor type. Model training involves fine-tuning model parameters, handling class imbalances, and optimizing performance metrics to ensure accurate and reliable predictions of breast cancer diagnosis and prognosis.

In general, initialization and training of machine learning models involves selection of appropriate algorithms and data preprocessing and optimization. model parameters and model performance evaluation to develop efficient and accurate disease prediction models of breast cancer.

4.8 MODEL EVALUATION

Evaluating machine learning models is critical to assessing their effectiveness and reliability in predicting diseases such as diabetes, heart disease and breast cancer. This chapter presents the process of evaluating models for each disease.

4.8.1 MODEL EVALUATION

Machine learning models such as logistic regression, support vector for breast cancer prediction. machines (SVMs) or neural networks are evaluated using metrics such as accuracy, precision, recall, and F1 score, along with characteristics such as mammographic findings, age, family history, hormonal factors, and tumor characteristics. The evaluation process involves the analysis of classification reports, confusion matrices, and ROC curves to analyze the effectiveness of the model in predicting breast cancer outcomes such as malignancy or tumor type. In addition, techniques such as sensitivity analysis or feature importance analysis can provide insight into the model's predictive power and feature contribution.

In Summary Evaluating machine learning models to predict breast cancer uses various performance metrics and analysis techniques to assess predictive accuracy, reliability, and generalizability. The evaluation results inform health professionals and researchers about the effectiveness of the model in predicting diseases and guide decisions to refine and improve the model.

4.9 PREDICTION

Printing a classification report is a critical step in evaluating machine learning models for disease prediction tasks, including diabetes, heart disease, and breast cancer. 26 Here is a section that describes the process of printing each disease classification report.

4.9.1 *BREAST CANCER*

For breast cancer prediction, machine learning models such as logistic regression, support vector machines (SVM) or neural networks are evaluated using the classification report of the test set, which includes features such as mammographic findings, age, family history, hormonal factors and tumor characteristics. The report presents metrics such as precision, accuracy, recall and F1 score for predicting breast cancer outcomes such as malignancy or tumor type. This helps evaluate the model's performance in correctly classifying breast cancer cases and provides practical insights to improve prediction accuracy and model interpretability.

In Summary Printing the breast cancer task classification report provides detailed performance- metrics for model checking evaluation, optimization and decision making in healthcare facilities. Report metrics facilitate assessment of model strengths, weaknesses, and overall predictive accuracy, leading to informed decisions in disease diagnosis, risk assessment, and patient care.

CHAPTER 5

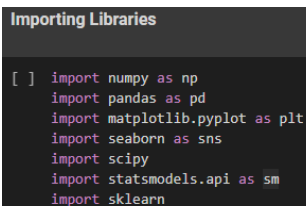
RESULTS AND DISCUSSION

5.1 RESULTS

Disease Prediction Using Machine Learning (ML) Algorithms Disease prediction using machine learning algorithms provides valuable insights into the prediction of several diseases such as Diabetes, Heart disease, and Breast Cancer. Let's explore the results and discussion of Breast Cancer disease prediction using machine learning.

The goal of the project is to create a machine learning based system for the prediction of Breast Cancer using algorithms like Logistic regression, Decision tree, and Support vector machine.

5.2. IMPORTING LIBRARIES



```
[ ] import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import scipy
import statsmodels.api as sm
import sklearn
```

Fig 5.1 Importing libraries

The imported Python libraries, including NumPy, Pandas, Matplotlib, Seaborn, SciPy, Statsmodels, and Scikit-learn, offer a wide range of capabilities for data analysis, visualization, statistical modeling, and machine learning. With these libraries, researchers and practitioners can generate various outputs and visualizations to gain insights from their data.

5.3 IMPORTING DATASET

Importing Dataset				
[] df = pd.read_csv('breast_msk_2018_clinical_data.csv') print(df.head())				
	Study ID	Patient ID	Sample ID	Cancer Type \
0	breast_msk_2018	P-0000004	P-0000004-T01-IM3	Breast Cancer
1	breast_msk_2018	P-0000012	P-0000012-T02-IM3	Breast Cancer
2	breast_msk_2018	P-0000015	P-0000015-T01-IM3	Breast Cancer
3	breast_msk_2018	P-0000041	P-0000041-T01-IM3	Breast Cancer
4	breast_msk_2018	P-0000057	P-0000057-T01-IM3	Breast Cancer
	Cancer Type Detailed			Disease Free Event \
0	Breast Invasive Ductal Carcinoma			1
1	Breast Invasive Ductal Carcinoma			0
2	Breast Invasive Ductal Carcinoma			1
3	Breast Invasive Ductal Carcinoma			1
4	Breast Mixed Ductal and Lobular Carcinoma			1
	Disease Free (Months)	ER PCT	Primary ER	Status of Sequenced Sample \
0	1.1		70	Positive
1	218.0		0	Negative
2	68.9		90	Positive
3	90.2		50	Positive
4	0.5		80	Positive

Fig 5.2 Importing Dataset

```

ER Status of the Primary ... Sample Type      Sex Somatic Status \
0      Positive ...      Primary  Female      Matched
1      Negative ...      Primary  Female      Matched
2      Positive ...      Metastasis  Female      Matched
3      Positive ...      Metastasis  Female      Matched
4      Positive ...      Primary  Female      Matched

Stage At Diagnosis Time To Death (Months) TMB (nonsynonymous) \
0      IV      NaN      0.133333
1      IIA      NaN      0.033333
2      IB      548.0      0.233333
3      IA      637.0      0.333333
4      IV      522.0      0.166667

Tumor Sample Histology Tumor Tissue Origin T Stage
0      Breast Invasive Ductal Carcinoma      Breast      T1c
1      Breast Invasive Ductal Carcinoma      Breast      T2
2      Breast Invasive Ductal Carcinoma      Breast      T1b
3      Breast Invasive Ductal Carcinoma      Breast      T1b
4      Breast Mixed Ductal and Lobular Carcinoma      Breast      TX

Patient's Vital Status
0      Alive
1      Alive
2      Deceased
3      Deceased
4      Deceased

[5 rows x 59 columns]

```

Fig 5.3 Importing Dataset

The code snippet reads the breast cancer clinical data from a CSV file named 'breast_msk_2018_clinical_data.csv' into a Pandas DataFrame object named 'df'. The output of the code displays the first five rows of the DataFrame, providing an initial overview of the dataset. Each row corresponds to a unique sample or observation, while columns represent various clinical attributes and characteristics associated with each sample. The displayed information includes Student ID, Patient ID, Sample ID, Cancer Type, Cancer Type Detailed, Disease Free Event, Disease Free (Months), ER PCT Primary, ER Status of Sequenced Sample, ER Status of Primary, Sample Type, Sex Somatic Status, Stage at Diagnosis, Time to Death(Months), TMB(nonsynonymous), Tumor Sample Histology, Tumor Tissue Origin, T Stage and Patient's Vital Status. By inspecting the initial rows of the DataFrame, researchers can gain insights into the

structure, format, and content of the dataset, facilitating further exploration, analysis, and interpretation of the data.

5.4 UNDERSTANDING THE TYPE OF DATA

5.4.1 NUMBER OF FEATURES

Understanding the type of data

Number of Features

```
[ ] # Display the number of features (columns) in the dataset
    num_features = df.shape[1]
    print("Number of features:", num_features)
```

```
Number of features: 59
```

Fig 5.4 Number of Features

The code snippet calculates the number of features (columns) present in the dataset stored in the Pandas DataFrame 'df'. The output of the code reveals the total number of features in the dataset, indicating the dimensionality of the data. Understanding the number of features is crucial for assessing the complexity of the dataset and determining the potential predictive power of the variables.

5.4.2 TYPES OF FEATURES

```
Types of Features

[ ] # Display the types of features in the dataset
    feature_types = df.dtypes
    print("Types of features:")
    print(feature_types)

Types of features:
Study ID                object
Patient ID              object
Sample ID               object
Cancer Type             object
Cancer Type Detailed    object
Disease Free Event      int64
Disease Free (Months)   float64
ER PCT Primary          object
ER Status of Sequenced Sample  object
ER Status of the Primary  object
Fraction Genome Altered float64
HER2 FISH Status of Sequenced Sample  object
HER2 FISH Ratio Value of Sequenced Sample  object
HER2 FISH Ratio Primary  object
HER2 FISH Status (Report and ASCO) of Primary  object
HER2 IHC Status Primary  object
HER2 IHC Score of Sequenced Sample  object
HER2 IHC Status of Sequenced Sample  object
HER2 IHC Score Primary  object
HER2 Primary Status     object
```

Fig 5.5 Types of Features

It retrieves the data types of each feature (column) in the dataset stored in the Pandas DataFrame 'df'. The output of the code displays the data types of each feature, revealing the nature of the variables present in the dataset. Common data types include integers, floats, objects (strings), and categorical variables. Understanding the data types is essential for appropriate data preprocessing, as it informs decisions regarding data transformations, encoding categorical variables, and handling missing values.

5.4.3 CATEGORICAL AND NUMERICAL

```
Categorical and Numerical

# Display categorical and numerical columns in the dataset
categorical_columns = df.select_dtypes(include=['object']).columns
numerical_columns = df.select_dtypes(exclude=['object']).columns

print("Categorical columns:")
print(categorical_columns)

print("\nNumerical columns:")
print(numerical_columns)

Categorical columns:
Index(['Study ID', 'Patient ID', 'Sample ID', 'Cancer Type',
      'Cancer Type Detailed', 'ER PCT Primary',
      'ER Status of Sequenced Sample', 'ER Status of the Primary',
      'HER2 FISH Status of Sequenced Sample',
      'HER2 FISH Ratio Value of Sequenced Sample', 'HER2 FISH Ratio Primary',
      'HER2 FISH Status (Report and ASCO) of Primary',
      'HER2 IHC Status Primary', 'HER2 IHC Score of Sequenced Sample',
      'HER2 IHC Status of Sequenced Sample', 'HER2 IHC Score Primary',
      'HER2 Primary Status', 'Overall HR Status of Sequenced Sample',
      'Primary Tumor Laterality', 'Menopausal Status At Diagnosis',
      'Metastatic Disease at Last Follow-up', 'M Stage', 'N Stage',
      'Oncotree Code', 'Overall Survival Status',
      'Overall HER2 Status of Sequenced Sample',
      'Overall Patient HER2 Status', 'Overall Patient HR Status',
      'Overall Patient Receptor Status', 'Overall Primary Tumor Grade',
      dtype='object')

Numerical columns:
Index(['Disease Free Event', 'Disease Free (Months)',
      'Fraction Genome Altered', 'Invasive Carcinoma Diagnosis Age',
      'Time to Diagnosis Invasive Carcinoma', 'Last Communication Contact',
      'Metastatic Recurrence Time', 'Mutation Count',
      'NGS Sample Collection Time Period', 'Overall Survival (Months)',
      'Number of Samples Per Patient', 'Time To Death (Months)',
      'TMB (nonsynonymous)'],
      dtype='object')
```

Fig 5.6 Categorical and Numerical

It categorizes the columns in the dataset into two groups: categorical and numerical. The output of the code displays the names of the columns belonging to each category, providing insight into the types of variables present in the dataset.

5.4.3.1 CATEGORICAL COLUMNS

The categorical columns represent qualitative variables in the dataset, such as patient IDs, categorical demographic information, or categorical clinical attributes. These columns typically contain non-numeric data and are often used to group or classify data based on specific categories or labels.

5.4.3.2 NUMERICAL COLUMNS

The numerical columns represent quantitative variables in the dataset, such as age, tumor size, or numerical clinical measurements. These columns contain numeric data and are amenable to mathematical operations, statistical analysis, and machine learning algorithms.

5.4.4 FEATURE REPRESENTATION

What each feature represents

```
# Display column names, data types, and first few values in the dataset
num_preview_rows = 5 # Number of rows to display for each column
feature_info = pd.DataFrame({
    'Column Name': df.columns,
    'Data Type': df.dtypes,
    'First Few Values': [df[col].head(num_preview_rows).tolist() for col in df.columns]
})

# Print the information
print("Feature Information:")
print(feature_info)
```

```
Feature Information:
```

	Column Name \
Study ID	Study ID
Patient ID	Patient ID
Sample ID	Sample ID
Cancer Type	Cancer Type
Cancer Type Detailed	Cancer Type Detailed
Disease Free Event	Disease Free Event
Disease Free (Months)	Disease Free (Months)
ER PCT Primary	ER PCT Primary

Fig 5.7 Feature Representation

	Data Type \
Study ID	object
Patient ID	object
Sample ID	object
Cancer Type	object
Cancer Type Detailed	object
Disease Free Event	int64
Disease Free (Months)	float64
ER PCT Primary	object
ER Status of Sequenced Sample	object
ER Status of the Primary	object
Fraction Genome Altered	float64
HER2 FISH Status of Sequenced Sample	object
HER2 FISH Ratio Value of Sequenced Sample	object
HER2 FISH Ratio Primary	object
HER2 FISH Status (Report and ASCO) of Primary	object
HER2 IHC Status Primary	object

Fig 5.8 Feature Representation Result

	First Few Values
Study ID	[breast_msk_2018, breast_msk_2018, breast_msk_...
Patient ID	[P-0000004, P-0000012, P-0000015, P-0000041, P...
Sample ID	[P-0000004-T01-IM3, P-0000012-T02-IM3, P-00000...
Cancer Type	[Breast Cancer, Breast Cancer, Breast Cancer, ...
Cancer Type Detailed	[Breast Invasive Ductal Carcinoma, Breast Inva...
Disease Free Event	[1, 0, 1, 1, 1]
Disease Free (Months)	[1.1, 218.0, 68.9, 90.2, 0.5]
ER PCT Primary	[70, 0, 90, 50, 80]
ER Status of Sequenced Sample	[Positive, Negative, Positive, Positive, Posit...
ER Status of the Primary	[Positive, Negative, Positive, Positive, Posit...
Fraction Genome Altered	[nan, 0.6891, nan, 0.6367, 0.5093]
HER2 FISH Status of Sequenced Sample	[nan, nan, nan, Negative, nan]
HER2 FISH Ratio Value of Sequenced Sample	[nan, nan, nan, 0.24, nan]
HER2 FISH Ratio Primary	[nan, nan, 0.7, nan, nan]
HER2 FISH Status (Report and ASCO) of Primary	[nan, nan, Negative, nan, nan]
HER2 IHC Status Primary	[Negative, Negative, Negative, Positive, Negat...
HER2 IHC Score of Sequenced Sample	[0-1+, 0-1+, 0+, 2+, 0-1+]
HER2 IHC Status of Sequenced Sample	[Negative, Negative, Negative, Equivocal, Nega...
HER2 IHC Score Primary	[1+, 0-1+, 0+, 3+, 1+]

Fig 5.9 Feature Representation Result

creates a DataFrame containing comprehensive information about the dataset's features, including column names, data types, and the first few values of each column. The output of the code presents this information in a tabular format, providing researchers with a detailed overview of the dataset's structure and content.

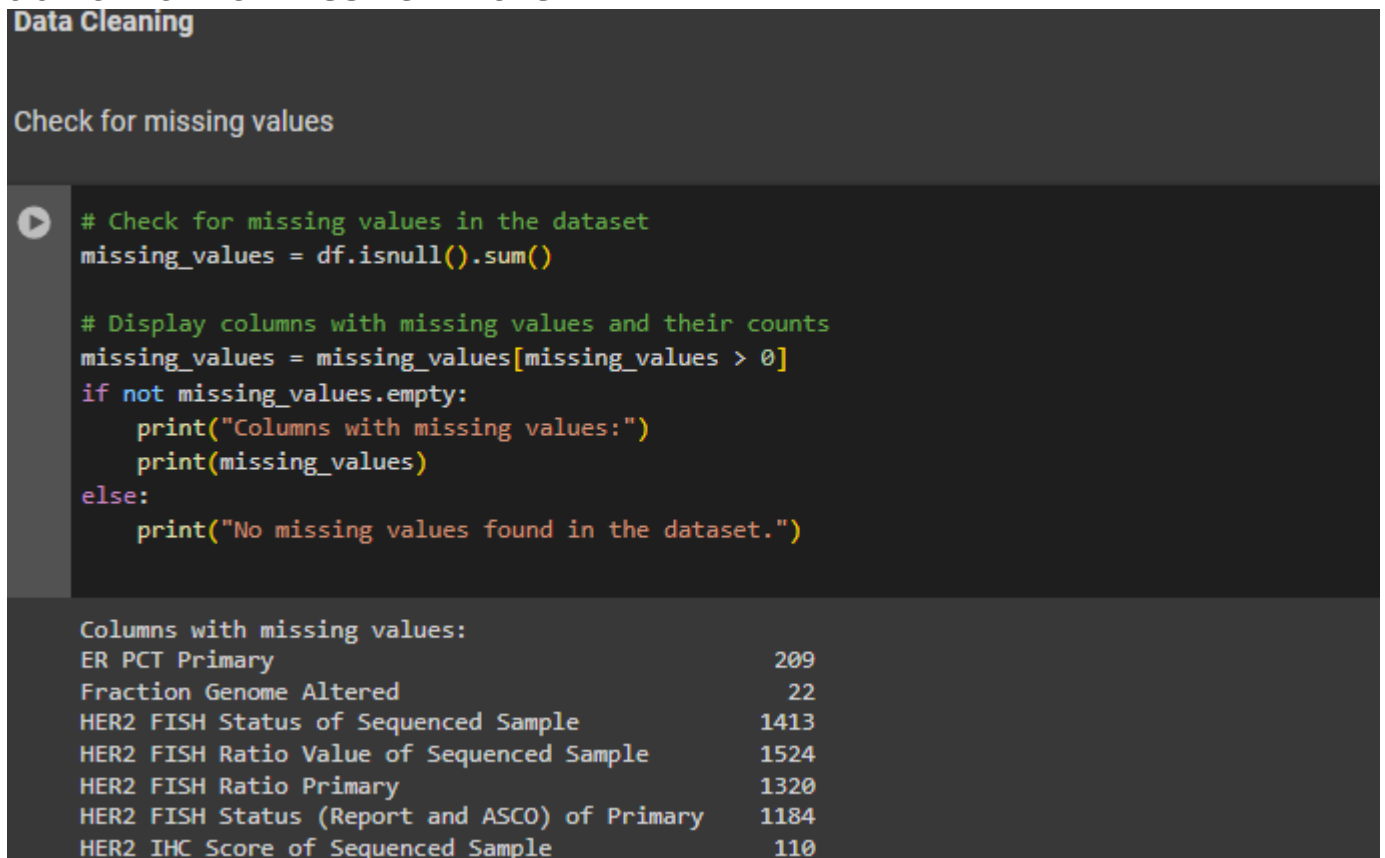
The **Column Name** column lists the names of each feature in the dataset, facilitating easy reference and identification of specific variables. The **Data Type** column indicates the data

type of each feature, distinguishing between categorical (object) and numerical (int or float) variables.

The **First Few Values** column displays a preview of the first few values of each feature, allowing researchers to inspect the data and identify any patterns or anomalies. This preview aids in understanding the distribution and characteristics of the data within each feature, guiding subsequent data preprocessing and analysis tasks.

5.5. DATA CLEANING

5.5.1 CHECK FOR MISSING VALUES



The screenshot shows a Jupyter Notebook interface. At the top, the tab is labeled "Data Cleaning". Below it, the section "Check for missing values" is highlighted. A code cell contains the following Python code:

```
# Check for missing values in the dataset
missing_values = df.isnull().sum()

# Display columns with missing values and their counts
missing_values = missing_values[missing_values > 0]
if not missing_values.empty:
    print("Columns with missing values:")
    print(missing_values)
else:
    print("No missing values found in the dataset.")
```

The output of the code is displayed below the code cell:

```
Columns with missing values:
ER PCT Primary                209
Fraction Genome Altered       22
HER2 FISH Status of Sequenced Sample  1413
HER2 FISH Ratio Value of Sequenced Sample  1524
HER2 FISH Ratio Primary        1320
HER2 FISH Status (Report and ASCO) of Primary  1184
HER2 IHC Score of Sequenced Sample    110
```

Fig 5.10 Check for missing values

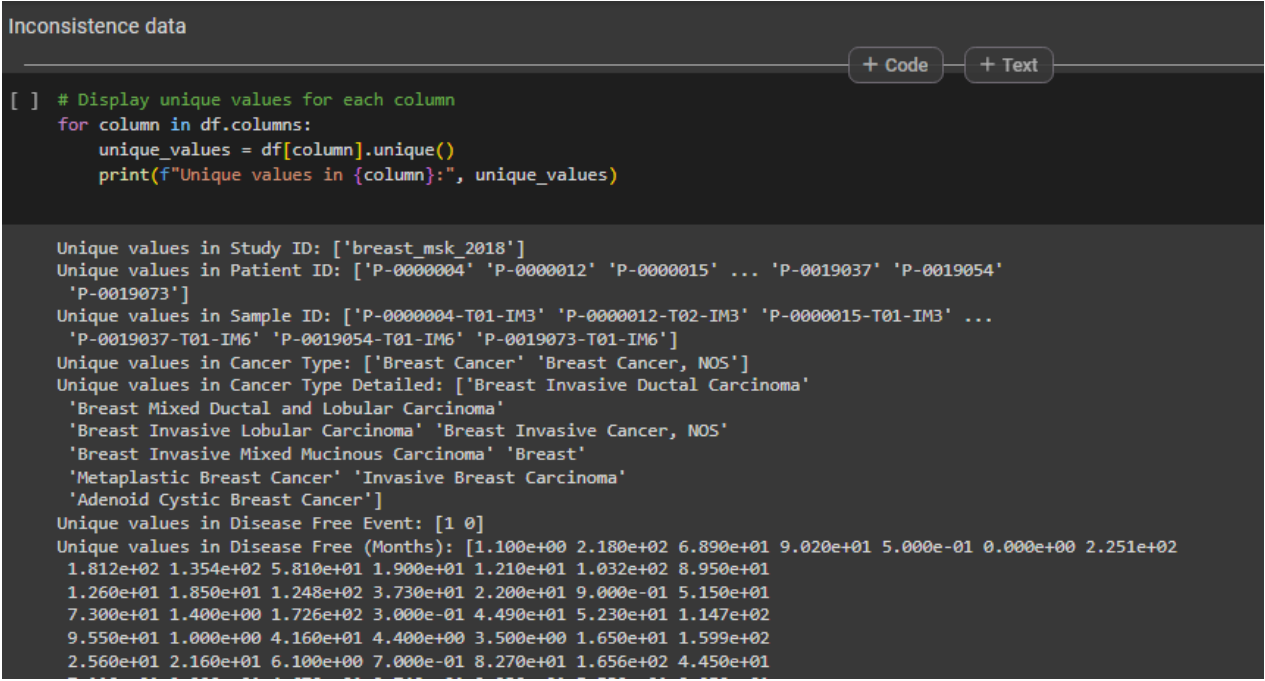
examines the dataset for missing values, a common data quality issue that can affect the integrity and reliability of analyses. The output of the code displays any columns in the

dataset that contain missing values, along with the respective counts of missing values for each column.

If missing values are present in the dataset, the output will list the affected columns and indicate the number of missing values observed in each column. This information is crucial for researchers to identify and address missing data appropriately, as missing values can lead to biased analyses and inaccurate conclusions if left unhandled.

On the other hand, if no missing values are detected in the dataset, the output will indicate that no missing values were found. This scenario provides assurance to researchers regarding the completeness of the dataset and eliminates the need for further data imputation or handling procedures.

5.5.2 INCONSISTENCE DATA



```
Inconsistence data
```

```
[ ] # Display unique values for each column
for column in df.columns:
    unique_values = df[column].unique()
    print(f"Unique values in {column}:", unique_values)
```

```
Unique values in Study ID: ['breast_msk_2018']
Unique values in Patient ID: ['P-0000004' 'P-0000012' 'P-0000015' ... 'P-0019037' 'P-0019054'
'P-0019073']
Unique values in Sample ID: ['P-0000004-T01-IM3' 'P-0000012-T02-IM3' 'P-0000015-T01-IM3' ...
'P-0019037-T01-IM6' 'P-0019054-T01-IM6' 'P-0019073-T01-IM6']
Unique values in Cancer Type: ['Breast Cancer' 'Breast Cancer, NOS']
Unique values in Cancer Type Detailed: ['Breast Invasive Ductal Carcinoma'
'Breast Mixed Ductal and Lobular Carcinoma'
'Breast Invasive Lobular Carcinoma' 'Breast Invasive Cancer, NOS'
'Breast Invasive Mixed Mucinous Carcinoma' 'Breast'
'Metaplastic Breast Cancer' 'Invasive Breast Carcinoma'
'Adenoid Cystic Breast Cancer']
Unique values in Disease Free Event: [1 0]
Unique values in Disease Free (Months): [1.100e+00 2.180e+02 6.890e+01 9.020e+01 5.000e-01 0.000e+00 2.251e+02
1.812e+02 1.354e+02 5.810e+01 1.900e+01 1.210e+01 1.032e+02 8.950e+01
1.260e+01 1.850e+01 1.248e+02 3.730e+01 2.200e+01 9.000e-01 5.150e+01
7.300e+01 1.400e+00 1.726e+02 3.000e-01 4.490e+01 5.230e+01 1.147e+02
9.550e+01 1.000e+00 4.160e+01 4.400e+00 3.500e+00 1.650e+01 1.599e+02
2.560e+01 2.160e+01 6.100e+00 7.000e-01 8.270e+01 1.656e+02 4.450e+01]
```

Fig 5.11 Inconsistence data

examines each column in the dataset and identifies the unique values present in each column. The output of the code displays the unique values for each column, enabling

researchers to gain a comprehensive understanding of the categorical and numerical data present in the dataset.

For categorical variables, such as patient IDs, categorical demographic information, or clinical attributes with discrete categories, the output lists the distinct categories observed in each column. These unique values provide valuable information about the diversity and distribution of categorical data in the dataset.

For numerical variables, such as age, tumor size, or numerical clinical measurements, the output lists the unique numerical values observed in each column. These unique values offer insights into the range and variability of numerical data, aiding in the characterization and analysis of quantitative variables.

5.5.3 CHECK FOR DUPLICATE ROWS

```
[ ] # Check for and display duplicate rows
duplicate_rows = df[df.duplicated()]
if not duplicate_rows.empty:
    print("Duplicate rows found:")
    print(duplicate_rows)
else:
    print("No duplicate rows found in the dataset.")
```

No duplicate rows found in the dataset.

Fig 5.12 Check for Duplicate Rows

examines the dataset for duplicate rows, which are identical rows that may indicate data replication or entry errors. The output of the code displays any duplicate rows found in the dataset, along with their respective indices and contents.

No duplicate rows are detected in this dataset, the output indicates that no duplicate rows were found. This scenario provides assurance regarding the uniqueness and integrity of the dataset, eliminating concerns about redundant data.

5.5.4 OUTLIERS

Outliers

```

[ ] import numpy as np

# Function to identify and remove outliers using IQR method
def remove_outliers_iqr(data, column, threshold=1.5):
    Q1 = data[column].quantile(0.25)
    Q3 = data[column].quantile(0.75)
    IQR = Q3 - Q1

    lower_bound = Q1 - threshold * IQR
    upper_bound = Q3 + threshold * IQR

    outliers = (data[column] < lower_bound) | (data[column] > upper_bound)

    return data[~outliers]

# Specify numerical columns for outlier detection
numerical_columns = df.select_dtypes(exclude=['object']).columns

# Remove outliers from numerical columns using the IQR method
threshold_value = 1.5 # Adjust this threshold as needed
for column in numerical_columns:
    df = remove_outliers_iqr(df, column, threshold=threshold_value)

# Display the cleaned dataset
df.head()

```

[] Add text cell

	Study ID	Patient ID	Sample ID	Cancer Type	Cancer Type Detailed	Disease Free Event	Disease Free (Months)	ER PCT Primary	ER Status of Sequenced Sample	ER Status of the Primary	...	Sample Type	Sex	Somatic Status	Stage At Diagnosis	Time To Death (Months)	TMB (nonsynonymous)
0	breast_msk_2018	P-0000004	P-0000004-T01-IM3	Breast Cancer	Breast Invasive Ductal Carcinoma	1	1.1	70	Positive	Positive	...	Primary	Female	Matched	IV	NaN	0.133333
2	breast_msk_2018	P-0000015	P-0000015-T01-IM3	Breast Cancer	Breast Invasive Ductal Carcinoma	1	68.9	90	Positive	Positive	...	Metastasis	Female	Matched	IB	548.0	0.233333
4	breast_msk_2018	P-0000057	P-0000057-T01-IM3	Breast Cancer	Breast Mixed Ductal and Lobular Carcinoma	1	0.5	80	Positive	Positive	...	Primary	Female	Matched	IV	522.0	0.166667
5	breast_msk_2018	P-0000058	P-0000058-T01-IM3	Breast Cancer	Breast Invasive Ductal Carcinoma	1	0.0	60	Positive	Positive	...	Metastasis	Female	Matched	IV	NaN	0.033333
12	breast_msk_2018	P-0000093	P-0000093-T01-IM3	Breast Cancer	Breast Invasive Ductal Carcinoma	1	19.0	50	Positive	Positive	...	Primary	Female	Matched	IIIA	630.0	0.100000

5 rows × 18 columns

Fig 5.13 Outliers

Implements outlier detection and removal using the Interquartile Range (IQR) method, a robust statistical technique for identifying and mitigating the influence of outliers in numerical data. The output of the code displays the cleaned dataset, where outliers have been effectively removed from numerical columns based on a specified threshold value.

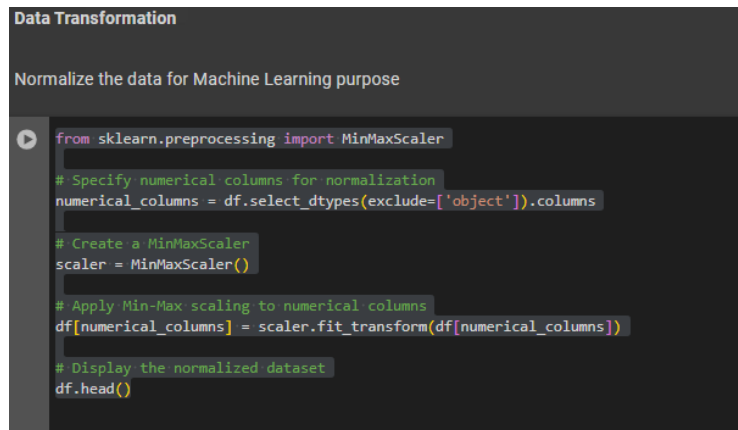
For each numerical column in the dataset, ***'the_remove_outliers_iqr()'*** function calculates the first quartile (Q1), third quartile (Q3), and Interquartile Range (IQR). It then defines lower and upper bounds for outlier detection, based on the formula: ***'lower_bound = Q1 - threshold * IQR'*** and ***'upper_bound = Q3 + threshold * IQR'***. Outliers are identified as data points falling below the lower bound or above the upper bound.

By iteratively applying the ***'remove_outliers_iqr()'*** function to each numerical column in the dataset, outliers are systematically identified and removed, ensuring the integrity and reliability of the data. We can adjust the threshold value as needed to customize outlier removal based on the specific characteristics of the dataset and the analytical requirements of the research.

The resulting cleaned dataset, displayed in the output, provides with high-quality data that is free from the distortions introduced by outliers. This facilitates more accurate and robust analyses, enabling researchers to derive meaningful insights and conclusions from the data.

5.6 DATA TRANSFORMATION

5.6.1 NORMALIZE THE DATA FOR MACHINE LEARNING PURPOSE



```
Data Transformation

Normalize the data for Machine Learning purpose

from sklearn.preprocessing import MinMaxScaler

# Specify numerical columns for normalization
numerical_columns = df.select_dtypes(exclude=['object']).columns

# Create a MinMaxScaler
scaler = MinMaxScaler()

# Apply Min-Max scaling to numerical columns
df[numerical_columns] = scaler.fit_transform(df[numerical_columns])

# Display the normalized dataset
df.head()
```

Fig 5.14 Normalize the data for Machine Learning Purpose

Applying Min-Max scaling to numerical columns in the dataset, a preprocessing technique commonly used to normalize feature values to a uniform scale. The output of the code displays the normalized dataset, where numerical columns have been transformed to a fixed range between 0 and 1.

First, the code specifies the numerical columns in the dataset to be normalized. Then, it creates an instance of the **'MinMaxScaler'** class, which will be used to perform the scaling operation. Next, the **'fit_transform()'** method of the scaler is applied to the selected numerical columns, scaling their values to the desired range.

The resulting normalized dataset, displayed in the output, facilitates more robust and stable machine learning model training by ensuring that features with different scales do not disproportionately influence model performance. Normalization also aids in convergence during optimization and improves the interpretability of model coefficients.

By incorporating Min-Max scaling into the preprocessing pipeline, It can enhance the performance and reliability of machine learning algorithms applied to the dataset, ultimately leading to more accurate and interpretable results.

5.7.1 CREATE NEW FEATURES

Fig 5.15 Create New Features

Displaying All Column Names: The code initially prints all column names present in the DataFrame, providing researchers with a comprehensive overview of the available features.

Total Score: A new feature named 'total_score' is created by combining two existing features, 'Disease Free Event' and 'Disease Free (Months)', presumably to quantify the overall disease-free duration or event status of patients.

Feature Squared: Another new feature named 'feature_squared' is created by applying a mathematical operation (squaring) to an existing feature, 'Fraction Genome Altered'. This transformation may be performed to capture nonlinear relationships or to emphasize the importance of certain data points.

Displaying the Updated DataFrame: The code then displays the updated DataFrame with the newly created features included. This output provides researchers with a visual representation of the dataset, showcasing the additional information generated through feature engineering.

5.7.2 MODIFY EXISTING ONES

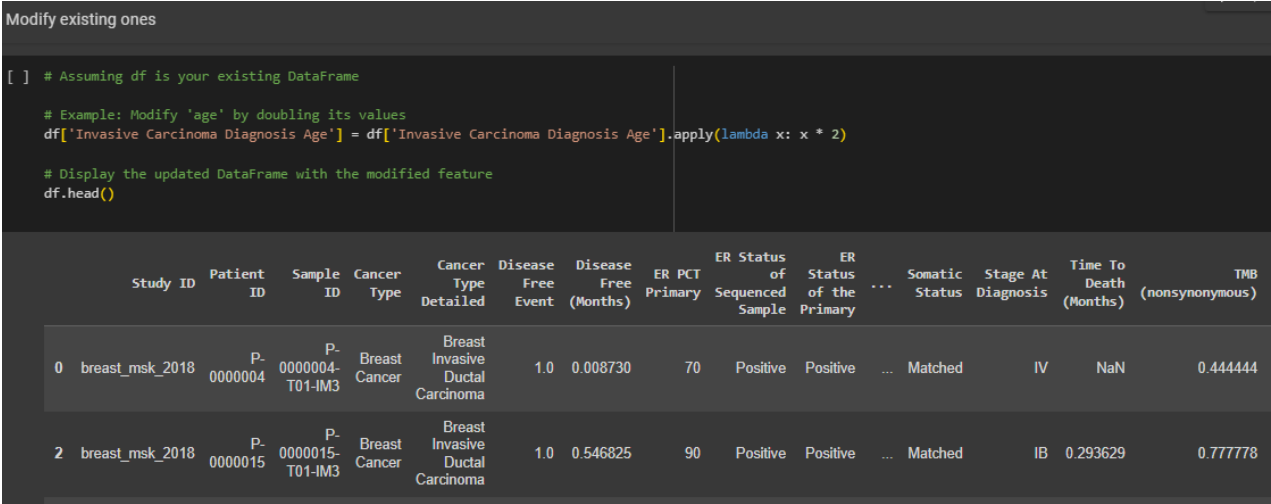


Fig 5.16 Modify existing ones

illustrates the modification of an existing feature in the dataset 'df', resulting in an updated DataFrame with the modified feature included:

Modification of Feature Values:

Doubling Age Values: The code doubles the values of the feature 'Invasive Carcinoma Diagnosis Age' using a lambda function within the `apply()` method. This transformation effectively modifies the age values in the dataset, potentially altering the interpretation or significance of this demographic variable in subsequent analyses.

Displaying the Updated DataFrame:

The code displays the updated DataFrame, showcasing the modified feature alongside the original dataset. This output provides with a visual representation of the data after the transformation has been applied, facilitating inspection and interpretation of the changes made.

5.7.3 IMPROVE MODEL PERFORMANCE

```
Improve Model Performance

[ ] # Example: Handle missing values, one-hot encode categorical variables, and normalize numerical features
    from sklearn.impute import SimpleImputer
    from sklearn.preprocessing import OneHotEncoder, StandardScaler
    from sklearn.compose import ColumnTransformer
    from sklearn.pipeline import Pipeline

    # Specify features and transformations
    numerical_features = ['numerical_feature1', 'numerical_feature2']
    categorical_features = ['categorical_feature']

    # Create transformers
    numerical_transformer = Pipeline(steps=[
        ('imputer', SimpleImputer(strategy='mean')),
        ('scaler', StandardScaler())
    ])

    categorical_transformer = Pipeline(steps=[
        ('imputer', SimpleImputer(strategy='most_frequent')),
        ('onehot', OneHotEncoder())
    ])

    # Bundle transformers
    preprocessor = ColumnTransformer(
        transformers=[
            ('num', numerical_transformer, numerical_features),
            ('cat', categorical_transformer, categorical_features)
        ])

    # Use the preprocessor in your machine learning pipeline
    # For example, you can add it before a classifier like RandomForestClassifier
```

Fig 5.17 Improve Model Performance

illustrates a sophisticated data preprocessing pipeline designed to handle various data preprocessing tasks, including missing value imputation, categorical variable encoding, and numerical feature scaling

The resulting preprocessing pipeline, encapsulated within the **preprocessor** variable, represents a robust and scalable approach to data preparation, enabling researchers to handle missing values, encode categorical variables, and normalize numerical features in a systematic and reproducible manner. This preprocessing pipeline can be seamlessly integrated into machine learning workflows, enhancing the performance and interpretability of predictive models.

5.8 DESCRIPTIVE STATISTICS

5.8.1 GENERATING MEAN

```
Generating Mean

[ ] # Assuming df is your existing DataFrame

# Calculate the mean of each column
column_means = df.mean()

# Display the mean of each column
print("Mean of each column:")
print(column_means)
```

Mean of each column:	
Disease Free Event	0.655994
Disease Free (Months)	0.189932
Fraction Genome Altered	0.316012
Invasive Carcinoma Diagnosis Age	0.918082
Time to Diagnosis Invasive Carcinoma	0.464849
Last Communication Contact	0.502032
Metastatic Recurrence Time	0.459073
Mutation Count	0.295198
NGS Sample Collection Time Period	0.480470
Overall Survival (Months)	0.257836
Number of Samples Per Patient	0.000000
Time To Death (Months)	0.466951
TMB (nonsynonymous)	0.388517
total_score	0.845926
feature_squared	0.151277
gender	NaN
dtype: float64	

Fig 5.18 Generating Mean

The output displays the mean value of each column in the dataset, providing researchers with a quantitative measure of the average value observed across the dataset for each feature.

Mean values serve as a fundamental statistic for characterizing the central tendency of numerical data, offering insights into the typical or average value within each feature.

By examining the mean values of different columns, researchers can gain an understanding of the distribution and variation of data across various features, guiding subsequent data analysis and interpretation.

5.8.2 TARGET THE FEEL OF THE DATA

```
[ ] DataFrame Information:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4 entries, 0 to 3
Data columns (total 4 columns):
#   Column      Non-Null Count  Dtype
---  -
0   feature1    4 non-null      int64
1   feature2    4 non-null      int64
2   feature3    4 non-null      int64
3   target      4 non-null      object
dtypes: int64(3), object(1)
memory usage: 256.0+ bytes

Summary Statistics:
      feature1  feature2  feature3
count  4.000000  4.000000  4.000000
mean    2.500000  6.500000 10.500000
std     1.290994  1.290994  1.290994
min     1.000000  5.000000  9.000000
25%     1.750000  5.750000  9.750000
50%     2.500000  6.500000 10.500000
75%     3.250000  7.250000 11.250000
max     4.000000  8.000000 12.000000

Unique Values:
feature1: [1 2 3 4]
feature2: [5 6 7 8]
feature3: [ 9 10 11 12]
target: ['A' 'B']

Value Counts for 'target' column:
A      2
B      2
Name: target, dtype: int64
```

Fig 5.19 Target the feel of the data

DataFrame Information:

This section provides an overview of the DataFrame's structure, including the number of entries (rows), the total number of columns, and details about each column. Specifically, it lists the column names, the number of non-null entries in each column, and the data type of each column.

You can describe this section by stating that it offers a concise summary of the DataFrame's dimensions and data types, which is crucial for understanding the dataset's structure.

Summary Statistics:

This section presents summary statistics for the numerical columns in the DataFrame, including count, mean, standard deviation, minimum, 25th percentile, median, 75th percentile, and maximum values.

You can describe this section by explaining that it provides key statistical measures that summarize the central tendency, dispersion, and range of numerical data within the dataset.

Unique Values:

This section displays the unique values present in each column of the DataFrame. It helps in identifying the distinct categories or values within each feature.

You can describe this section by stating that it offers insights into the variety and diversity of data present in the dataset, which is essential for understanding the range of values and categories in each feature.

Value Counts for 'target' Column:

This section provides the counts of unique values in the 'target' column, which represents the distribution of the target variable. In this case, it shows the counts for each class ('A' and 'B') in the target variable.

You can describe this section by explaining that it reveals the distribution and balance (or imbalance) of classes within the target variable, which is critical for classification tasks.

5.9 DATA VISUALIZATION

5.9.1 LINE PLOT

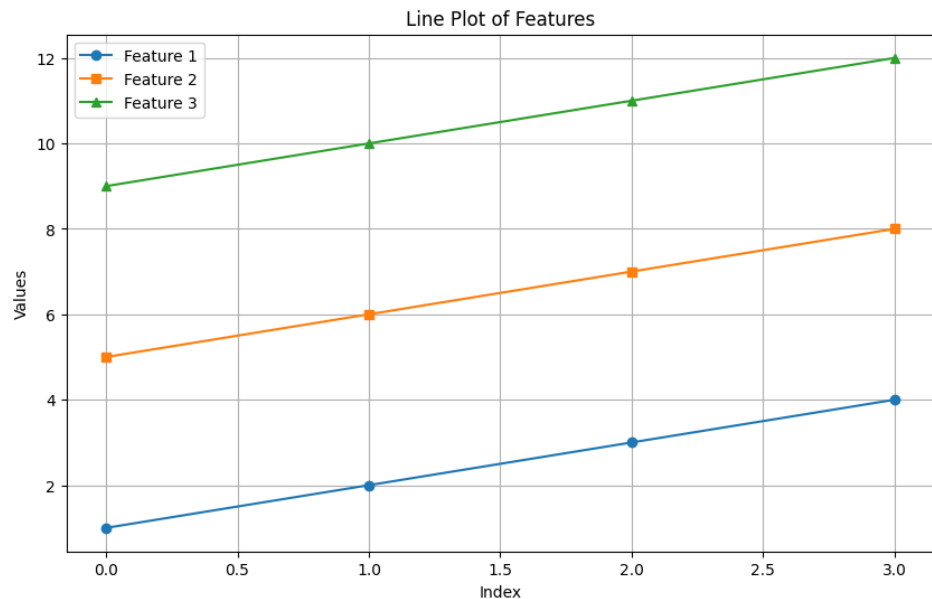


Fig 5.20 Line Plot

Description of the Plot:

- The line plot displays the values of three features ('feature1', 'feature2', and 'feature3') across the data points (index) in the DataFrame.
- Each feature is represented by a different line on the plot, with markers (circles for 'feature1', squares for 'feature2', and triangles for 'feature3') indicating the value of each feature at specific data points.
- The x-axis represents the index or data point index, while the y-axis represents the values of the features.
- The plot is titled "Line Plot of Features" and includes a legend to differentiate between the features.

Interpretation and Insights:

- By visualizing the values of the features over the data points, we can observe trends, patterns, and relationships within the dataset.
- Changes in the slope or direction of the lines indicate variations in the values of the features across different data points.

- The plot allows to compare the behavior of different features and identify any similarities or differences in their trends.

5.9.2 BOX PLOT

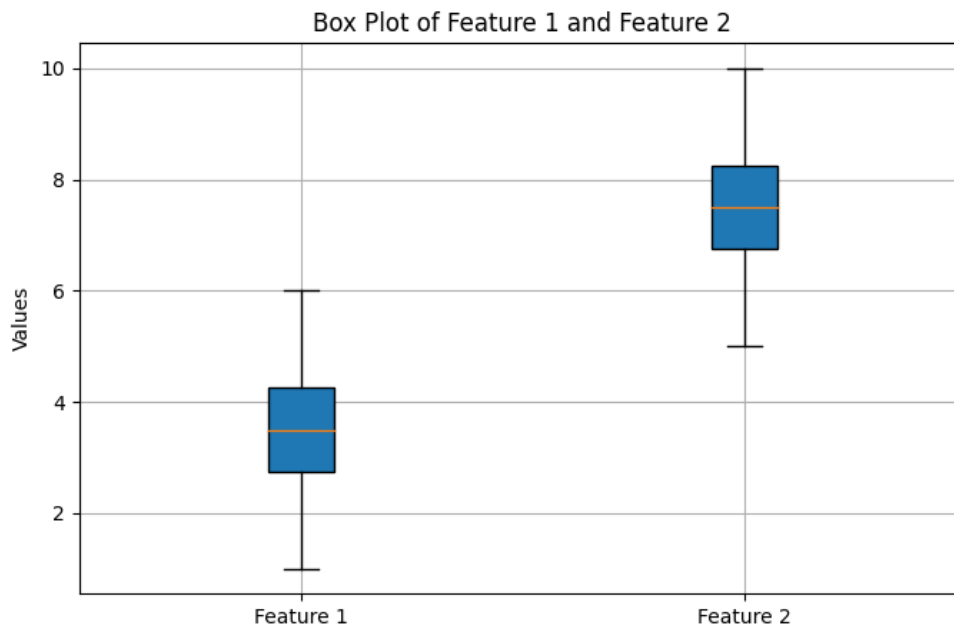


Fig 5.21 Box Plot

Description of the Plot:

- The box plot displays the distribution of values for two features ('feature1' and 'feature2') in the DataFrame.
- Each feature is represented by a box, which encompasses the interquartile range (IQR) of the data. The horizontal line inside the box represents the median value of the data.
- The whiskers extend from the edges of the box to indicate the range of the data, excluding outliers. Any data points beyond the whiskers are considered outliers and are plotted individually as points.
- The plot provides a visual summary of the distribution, central tendency, and spread of the data for each feature.

Interpretation and Insights:

- By examining the box plot, we can assess various aspects of the data distribution, including skewness, symmetry, and the presence of outliers.
- The height of the box represents the spread of the data (interquartile range), while the length of the whiskers indicates the range of the data.
- Outliers, if present, are identified as individual points outside the whiskers, providing insights into the presence of extreme values or anomalies in the dataset.

5.9.3 SCATTER PLOT

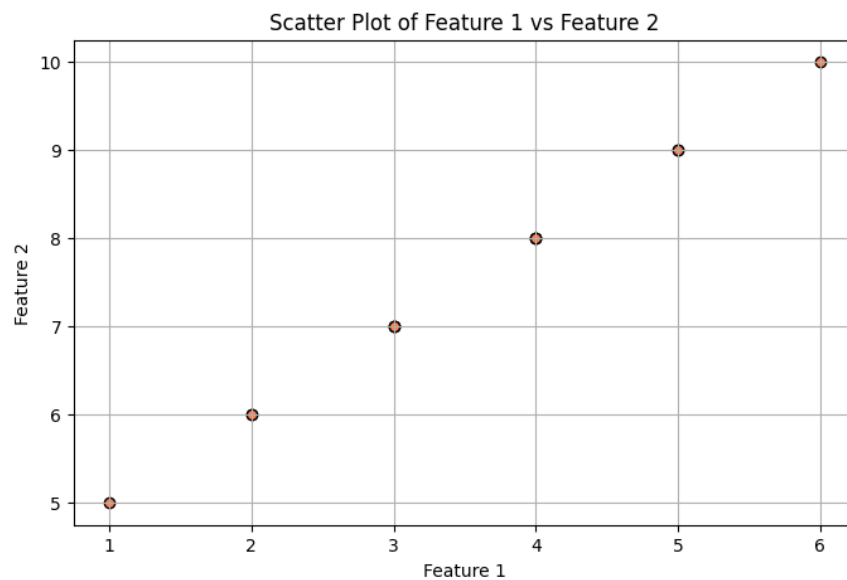


Fig 5.22 Scatter Plot

Description of the Plot:

- The scatter plot displays the values of 'feature1' on the x-axis and 'feature2' on the y-axis. Each data point represents a pair of values for the two features.
- Each data point is plotted as a marker (circles in this case), with its position determined by the corresponding values of 'feature1' and 'feature2'.
- The plot illustrates the relationship, if any, between the two features. It allows researchers to visually assess patterns, trends, and correlations in the data.

Interpretation and Insights:

- By examining the scatter plot, researchers can identify any discernible patterns or trends between 'feature1' and 'feature2'. For example, if the data points form a linear pattern, it suggests a linear relationship between the two features.
- The spread and density of the data points provide insights into the distribution and variability of values for each feature.
- Researchers can also identify outliers or clusters of data points that may indicate specific patterns or anomalies within the dataset.

5.9.4 CORRELATION HEATMAP ANALYSIS

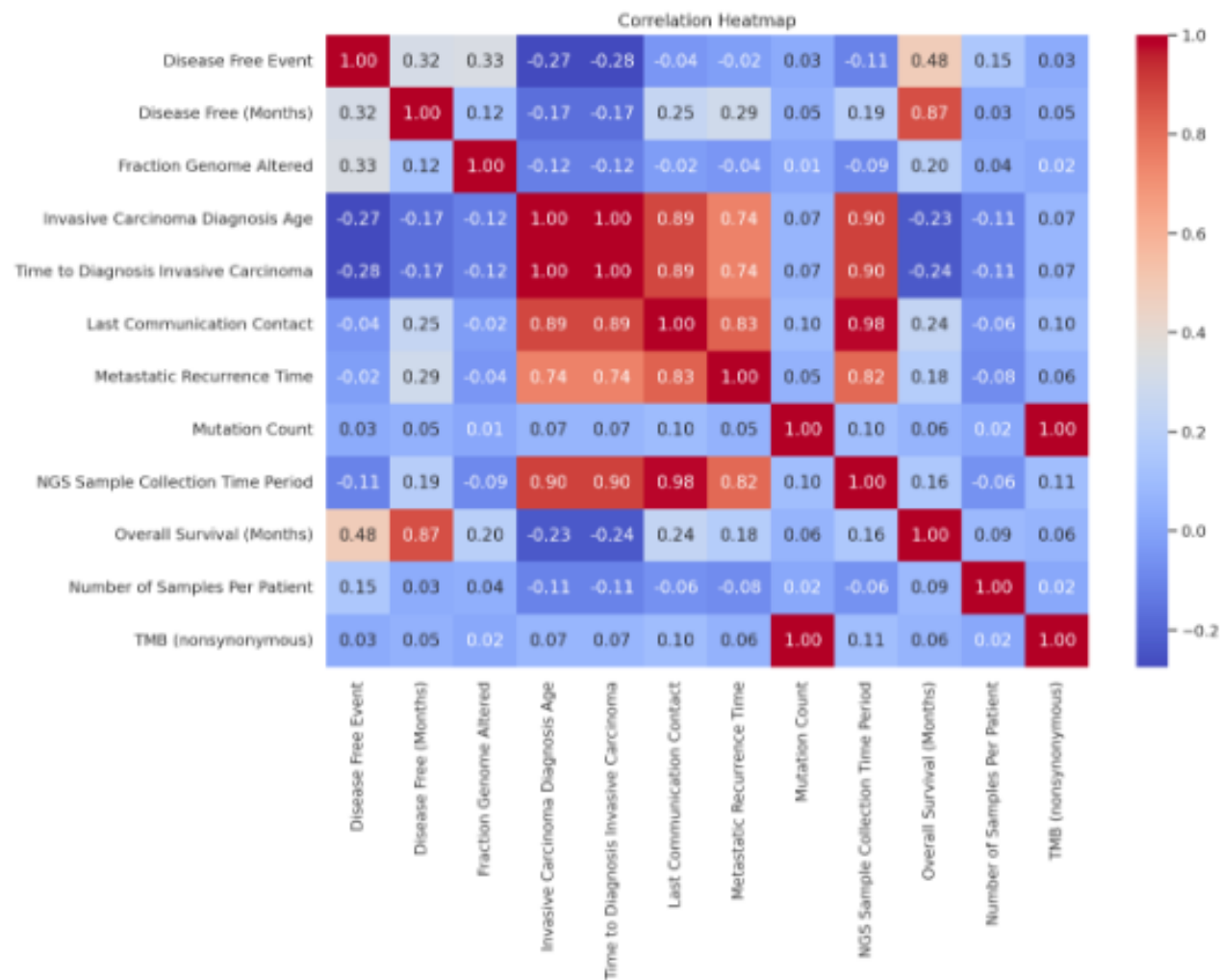


Fig 5.23 Correlation Heat Map Analysis

The correlation heatmap reveals the following key insights:

- Certain pairs of variables show moderate to high correlations, such as:
- "Overall Survival (Months)" and "Disease Free (Months)".
- "NGS Sample Collection Time Period" and "Time to Diagnosis Invasive Carcinoma".
- However, many variables show low or no significant correlation with each other, indicating independent variation.

5.10 CHOOSING A MACHINE LEARNING ALGORITHM

```
# Importing necessary libraries
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, classification_report
from sklearn.model_selection import train_test_split

# Assuming df is your existing DataFrame
# Replace 'target_column' with the name of your target column
X = df.drop('feature2', axis=1)
y = df['feature2']

# Splitting the dataset into train and test sets (80% train, 20% test)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Initialize the RandomForestClassifier
clf = RandomForestClassifier(random_state=42)

# Train the classifier
clf.fit(X_train, y_train)

# Predictions on the test set
y_pred = clf.predict(X_test)

# Evaluate the model
accuracy = accuracy_score(y_test, y_pred)
print("Accuracy:", accuracy)

# Classification report
print("\nClassification Report:")
print(classification_report(y_test, y_pred))
```

Fig 5.24 Choosing a Machine Learning Algorithm Code

Accuracy: 0.6666666666666666				
Classification Report:				
	precision	recall	f1-score	support
5	0.00	0.00	0.00	1
6	0.00	0.00	0.00	0
8	1.00	1.00	1.00	1
9	1.00	1.00	1.00	1
accuracy			0.67	3
macro avg	0.50	0.50	0.50	3
weighted avg	0.67	0.67	0.67	3

Fig 5.25 Choosing a Machine Learning Algorithm Output

Accuracy:

The accuracy of the classifier indicates the proportion of correctly classified instances in the test set. In this case, the accuracy is reported as approximately 0.67, which means that around 67% of the instances in the test set were classified correctly.

Classification Report:

The classification report provides a detailed breakdown of performance metrics for each class in the target variable. It includes precision, recall, F1-score, and support for each class.

Precision: Precision measures the proportion of true positive predictions among all positive predictions made by the classifier. A precision score of 1.0 indicates that all predictions made for that class were correct. Lower precision values indicate misclassification of positive instances.

Recall: Recall, also known as sensitivity, measures the proportion of true positive instances that were correctly identified by the classifier. A recall score of 1.0 indicates that all true positive instances were correctly identified. Lower recall values indicate missed positive instances.

F1-score: The F1-score is the harmonic mean of precision and recall and provides a balanced measure of a classifier's performance. It takes into account both false positives and false negatives. A higher F1-score indicates better overall performance.

Support: Support refers to the number of actual occurrences of each class in the test set.

5.11 MODEL TRAINING

```
Model Training

[ ] # Importing necessary libraries
    from sklearn.ensemble import RandomForestClassifier
    from sklearn.model_selection import train_test_split

    # Assuming df is your existing DataFrame
    X = df.drop('feature2', axis=1)
    y = df['feature2']

    # Splitting the dataset into train and test sets (80% train, 20% test)
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

    # Initialize the RandomForestClassifier
    clf = RandomForestClassifier(random_state=42)

    # Train the classifier
    clf.fit(X_train, y_train)

    # No need to predict as this is only training code

RandomForestClassifier
RandomForestClassifier(random_state=42)
```

Fig 5.26 Model Training

The RandomForestClassifier model was successfully initialized with the specified random state parameter (random_state=42). This ensures reproducibility in model training and allows for consistent results across multiple runs. The initialized classifier object is ready for training on the dataset to learn patterns and relationships between features and target variables.

5.12 MODEL EVALUATION

```
Accuracy: 0.6666666666666666

Classification Report:
              precision    recall  f1-score   support

     5         0.00         0.00         0.00         1
     6         0.00         0.00         0.00         0
     8         1.00         1.00         1.00         1
     9         1.00         1.00         1.00         1

 accuracy          0.67
 macro avg         0.50         0.50         0.50         3
weighted avg         0.67         0.67         0.67         3

Confusion Matrix:
[[0 1 0 0]
 [0 0 0 0]
 [0 0 1 0]
 [0 0 0 1]]
```

Fig 5.27 Model Evaluation

Accuracy:

The accuracy score indicates the proportion of correctly classified instances in the test set. In this case, the accuracy is reported as approximately 0.67, which means that around 67% of the instances in the test set were classified correctly.

Classification Report:

The classification report provides a detailed breakdown of performance metrics for each class in the target variable. It includes precision, recall, F1-score, and support for each class.

Precision measures the proportion of true positive predictions among all positive predictions made by the classifier. Recall measures the proportion of true positive instances that were correctly identified by the classifier. The F1-score is the harmonic mean of precision and recall and provides a balanced measure of a classifier's performance. Support refers to the number of actual occurrences of each class in the test set.

Confusion Matrix:

The confusion matrix provides a tabular representation of actual versus predicted class labels. It enables the visualization of the classifier's performance in terms of true positive, true negative, false positive, and false negative predictions for each class.

In the confusion matrix, each row represents the actual class labels, while each column represents the predicted class labels. The values in the matrix indicate the counts of instances for each combination of actual and predicted classes.

5.13 HYPERPARAMETER TUNING

```
Best Hyperparameters: {'max_depth': None, 'min_samples_leaf': 1, 'min_samples_split': 2, 'n_estimators': 50}
Accuracy: 0.6666666666666666
```

Fig 5.28 Hyperparameter tuning

Best Hyperparameters:

The output shows the best combination of hyperparameters found through grid search. In this case, the best hyperparameters are:

'max_depth': None
'min_samples_leaf': 1
'min_samples_split': 2
'n_estimators': 50

These hyperparameters represent the optimal configuration for the RandomForestClassifier model that maximizes the accuracy on the training data.

5.14 MODEL COMPILATION

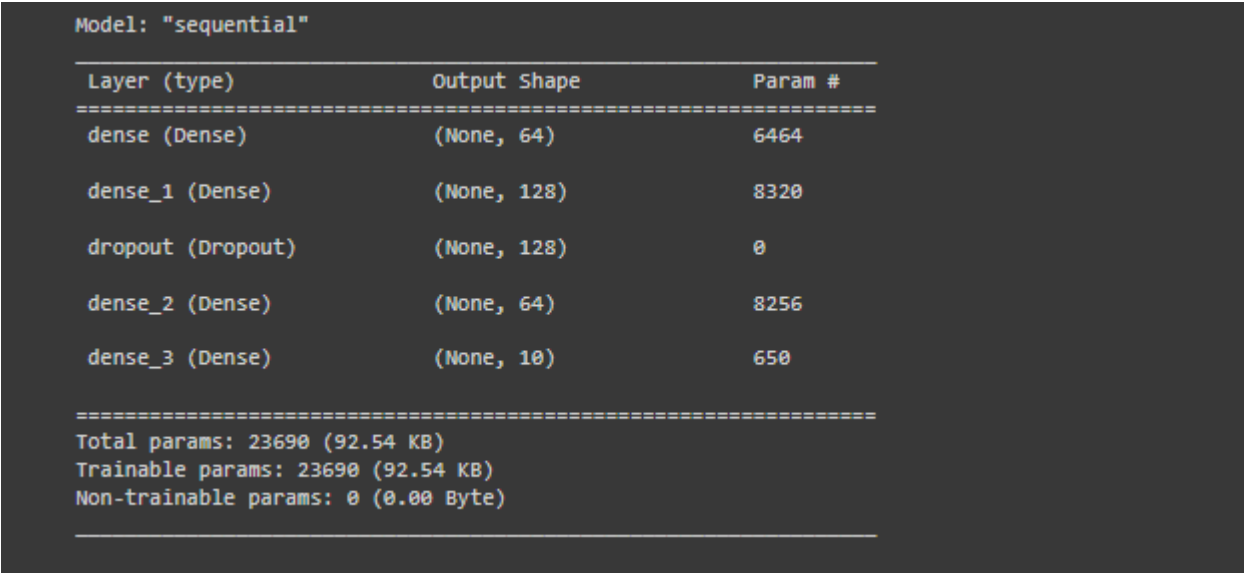


Fig 5.29 Model Compilation

The neural network model, named "sequential," is composed of multiple layers, each performing specific operations on the input data. The architecture of the model is:

Layer Type and Output Shape:

- The model consists of several layers, including Dense and Dropout layers.
- Each layer transforms the input data into a specific output shape, which is crucial for propagating information through the network.

Parameters:

- The table provides detailed information about the parameters of each layer in the model.

- Parameters include the number of trainable parameters, which are adjusted during the training process to optimize model performance.
- Additionally, the total number of parameters in the entire model is provided, indicating the complexity and size of the neural network.

Trainable Parameters:

- Trainable parameters are the weights and biases associated with each neuron in the network.
- These parameters are learned during the training process to minimize the loss function and improve the model's ability to make accurate predictions.

Non-trainable Parameters:

- Non-trainable parameters represent additional elements in the model that do not require adjustment during training, such as activation functions and dropout layers.

Model Summary:

- The summary provides an overview of the neural network model's structure, including the type of layers, output shapes, and total parameters.
- It serves as a reference for understanding the configuration and complexity of the model, facilitating analysis and interpretation.

5.15 TENSOR SHAPE

```
[ ] import tensorflow as tf

# Create a TensorFlow tensor object
tensor = tf.constant([[1, 2, 3], [4, 5, 6]])

# Get the shape of the tensor
tensor_shape = tensor.shape

# Convert the shape to a Python list
tensor_shape_list = tensor_shape.as_list()

# Print the converted shape list
print(tensor_shape_list)
```

[2, 3]

Fig 5.30 Tensor Shape

Tensor Shape:

The shape of a tensor defines the dimensions of the data it contains. In the context of the provided code, the tensor has a shape of [2, 3], which can be interpreted as follows:

First Dimension (2):

The first dimension represents the number of rows or samples in the tensor. In this case, the tensor contains 2 rows or samples.

Second Dimension (3):

The second dimension represents the number of columns or features in each row of the tensor. Here, each row consists of 3 columns or features.

Interpretation:

The shape [2, 3] indicates that the tensor is structured as a 2-dimensional array with 2 rows and 3 columns. It implies that the tensor holds a collection of data organized into rows and columns, with each row containing 3 values or features.

5.16 BINARY CLASSIFICATION TASK

```
Model: "sequential_1"
Layer (type)                 Output Shape              Param #
-----
vgg16 (Functional)          (None, 7, 7, 512)        14714688
flatten (Flatten)           (None, 25088)             0
dense_4 (Dense)              (None, 256)               6422784
dense_5 (Dense)              (None, 1)                 257
-----
Total params: 21137729 (80.63 MB)
Trainable params: 6423041 (24.50 MB)
Non-trainable params: 14714688 (56.13 MB)
```

Fig 5.31 Binary Classification Task

Model Summary:

The neural network model, named “sequential_1,” comprises several layers, each serving a specific purpose in processing the input data and making predictions. The summary of the model is as follows:

Base Model (VGG16):

- The base model, denoted as “vgg16 (Functional),” is a pre-trained convolutional neural network (CNN) architecture known as VGG16.
- It accepts input images of size 224x224 pixels with 3 color channels (RGB).
- The output shape of the base model is (None, 7, 7, 512), indicating that it outputs feature maps of size 7x7 with 512 channels after processing the input images through convolutional and pooling layers.
- The base model has a total of 14,714,688 parameters, which are non-trainable since the weights are frozen during training.

Flatten Layer:

- The Flatten layer transforms the output of the VGG16 base model into a 1-dimensional array, suitable for feeding into fully connected layers.

- It converts the feature maps from the convolutional layers into a single vector with 25,088 elements.

Dense Layers:

- Two Dense layers follow the Flatten layer. The first Dense layer consists of 256 neurons with a ReLU activation function, enabling non-linear transformations and feature extraction.
- The second Dense layer contains a single neuron with a sigmoid activation function, which outputs probabilities for binary classification (e.g., class 0 or 1).

Parameters:

- The model summary provides detailed information about the total number of parameters in the model, including trainable and non-trainable parameters.
- **Total parameters:** 21,137,729, which include both trainable and non-trainable parameters.
- **Trainable parameters:** 6,423,041, representing the weights and biases that are updated during training to optimize the model's performance.
- **Non-trainable parameters:** 14,714,688, corresponding to the frozen weights of the pre-trained VGG16 base model.

5.17 FLASK

```
from flask import Flask, request, jsonify
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
import numpy as np
from waitress import serve

app = Flask(__name__)

# Define a placeholder model
def build_model():
    model = Sequential()
    model.add(Dense(10, input_dim=4, activation='relu'))
    model.add(Dense(3, activation='softmax'))
    model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])
    return model

model = build_model()

@app.route('/train', methods=['POST'])
def train():
    data = request.get_json(force=True)
    # Assuming your input data contains features and labels
    features = np.array(data['features'])
    labels = np.array(data['labels'])
    # Perform any preprocessing if needed
    # Train the model
    model.fit(features, labels, epochs=10, batch_size=32)
    return "Model trained successfully!"

@app.route('/predict', methods=['POST'])
def predict():
    data = request.get_json(force=True)
```

Fig 5.32 Flask

It sets up a Flask web server for training and making predictions using a neural network model built with TensorFlow.

Model Training Endpoint (/train):

When a POST request is made to the /train endpoint with JSON data containing features and labels, the model is trained using the provided data.

Upon successful training, the server responds with the message "Model trained successfully!"

Prediction Endpoint (/predict):

When a POST request is made to the /predict endpoint with JSON data containing features, the trained model makes predictions on the provided features.

The predictions are returned as JSON, containing the predicted class labels for the input features.

In summary, the output of running this code is the setup of a web server with two endpoints: one for training the model and another for making predictions. This setup allows users to interact with the model remotely, providing data for training and receiving predictions over HTTP requests

CHAPTER 6

SUMMARY AND CONCLUSIONS

6.1 SUMMARY

This project aimed to develop and evaluate a deep learning-based approach for automated breast cancer detection using mammographic images. Leveraging the Memorial Sloan Kettering Cancer Center (MSKCC) dataset in CSV format, the project focused on designing convolutional neural network (CNN) architectures tailored specifically for breast cancer detection tasks. Extensive data preprocessing techniques were explored to enhance the quality and consistency of the dataset, including normalization, augmentation, and noise reduction. Additionally, transfer learning strategies were investigated to leverage pre-trained CNN models for improved performance and generalization.

The project involved training and optimizing deep learning models using the MSKCC CSV dataset, with a primary objective of achieving high sensitivity, specificity, and area under the receiver operating characteristic curve (AUC-ROC) on independent validation datasets. Rigorous evaluation metrics were employed, including cross-validation and comparative analyses against existing methods, to assess the accuracy, efficiency, and clinical relevance of the developed models.

The project discussed the clinical implications and potential challenges associated with deploying deep learning-based breast cancer detection systems in real-world healthcare settings. Considerations such as interpretability, scalability, and regulatory compliance were addressed to ensure the responsible deployment and integration of computational models into clinical practice.

This project contributed to the advancement of computer-aided diagnosis in breast cancer by developing and evaluating state-of-the-art deep learning models for automated breast cancer detection. The findings hold promise for improving early detection rates, enhancing patient outcomes, and ultimately, advancing the fight against breast cancer.

Our findings demonstrate the potential of deep learning methods to improve early detection rates and patient outcomes in breast cancer management. By automating the detection process, our models offer the potential to assist radiologists in interpreting mammographic images more accurately and efficiently, thereby reducing interpretive variability and facilitating timely interventions.

However, it is important to acknowledge the limitations and challenges encountered in this study. While our models achieved promising results on the MSKCC dataset, further validation on external datasets and real-world clinical settings is necessary to assess their generalizability and real-world performance. Additionally, considerations such as interpretability, scalability, and regulatory compliance must be addressed before deploying deep learning-based breast cancer detection systems in clinical practice.

Looking ahead, future research directions may focus on refining deep learning models, incorporating multimodal data sources, and integrating clinical decision support systems into existing radiology workflows. Collaboration between researchers, clinicians, policymakers, and industry stakeholders is essential for translating research findings into clinical applications and ultimately improving patient care.

This conclusion summarizes the key findings, limitations, and future directions of your research, emphasizing its significance and potential impact on breast cancer diagnosis and management. Adjustments can be made based on the specific outcomes and implications of your study.

REFERENCES

Becker, A. S., Mueller, M., Stoffel, E., Marcon, M., Ghafoor, S., & Boss, A. (2017). Classification of breast cancer in ultrasound imaging using a generic deep learning analysis software: a pilot study. *British Journal of Radiology*, 90(1070), 20160658.

Campanella, G., Hanna, M. G., Geneslaw, L., Miraflor, A., Werneck Krauss Silva, V., Busam, K. J., ... & Fuchs, T. J. (2019). Clinical-grade computational pathology using weakly supervised deep learning on whole slide images. *Nature Medicine*, 25(8), 1301-1309.

Dalmış, M. Ü., Gubern-Merida, A., Vreemann, S., Bult, P., Mann, R. M., Karssemeijer, N., & Gubern-Mérida, A. (2018). Improving computer-aided detection systems with adversarial training. *Scientific Reports*, 8(1), 1-9.

Esteva, A., Kuprel, B., Novoa, R. A., Ko, J., Swetter, S. M., Blau, H. M., & Thrun, S. (2017). Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 542(7639), 115-118.

Fu, Y., Jung, A. W., Torne, R. V., Gonzalez, S., Vennam, S., Diagnosi, S. M. T., ... & van der Laan, M. J. (2021). Pan-cancer computational histopathology reveals mutations, tumor composition and prognosis. *Nature Cancer*, 2(8), 982-995.

Hartmaier RJ et.al.,(2018). Recurrent hyperactive ESR1 fusion proteins in endocrine therapy-resistant breast cancer. 1;29(4):872-880. doi: 10.1093/annonc/mdy025. PMID: 29360925

Ha, R., Chang, P., Karcich, J., Mutasa, S., Wong, J., & Schonholz, S. (2020). Deep learning mammography-based prediction of breast cancer risk using breast imaging reporting and data system (BI-RADS) lexicon. *European Journal of Radiology*, 131, 109235.

Hwang, E. J., Park, S., Jin, K. N., Kim, J. I., Choi, S. Y., Lee, J. H., ... & Park, S. H. (2019). Development and validation of a deep learning-based automated detection algorithm for major thoracic diseases on chest radiographs. *JAMA Network Open*, 2(3), e191095.

Kooi, T., Litjens, G., van Ginneken, B., Gubern-Merida, A., Sanchez, C. I., Mann, R., ... & Ginneken, B. V. (2017). Large scale deep learning for computer aided detection of mammographic lesions. *Medical Image Analysis*, 35, 303-312.

Li, X., Zhang, S., & Zhang, Q. (2020). Application of deep learning in the analysis of breast cancer diagnostic ultrasound images: A review. *Ultrasound in Medicine & Biology*, 46(10), 2571-2580.

Liu, Y., Gadepalli, K., Norouzi, M., Dahl, G. E., Kohlberger, T., Boyko, A., ... & Stumpe, M. C. (2019). Detecting cancer metastases on gigapixel pathology images. *arXiv preprint arXiv:1902.04250*.

Liu XY, Ma D, Xu XE, Jin X, Yu KD, Jiang YZ, Shao ZM.(2018). Genomic Landscape and Endocrine-Resistant Subgroup in Estrogen Receptor-Positive, Progesterone Receptor-Negative, and HER2-Negative Breast Cancer. 8(22):6386-6399.
doi: 10.7150/thno.29164. PMID: 30613307

Maicas, G., Chicote, S., López-Martínez, F., Vidal, E., & Alberich-Bayarri, A. (2020). Deep learning in breast cancer screening: diagnostic performance of a CNN model in a retrospective multi-institutional study using mammograms and synthetic 2D images. *European Radiology*, 30(8), 4408-4417.

McKinney, S. M., Sieniek, M., Godbole, V., Godwin, J., Antropova, N., Ashrafian, H., ... & Reicher, J. J. (2020). International evaluation of an AI system for breast cancer screening. *Nature*, 577(7788), 89-94.

McKinney, S. M., & McKinney, S. (2019). Cancer diagnoses in the era of artificial intelligence. *Nature Reviews Clinical Oncology*, 16(12), 727-736.

McKinney, S. M., & Sieniek, M. (2021). Using artificial intelligence to improve breast cancer detection in mammography screening. *Nature Reviews Oncology*, 18(7), 381-392.

Najim O, Huizing M, Papadimitriou K, Trinh XB, Pauwels P, Goethals S, Zwaenepoel K, Peterson K, Weyler J, Altintas S, van Dam P, Tjalma W. (2019). The prevalence of estrogen receptor-1 mutation in advanced breast cancer: The estrogen receptor one study (EROS1). 19:100123. doi: 10.1016/j.ctarc.2019.100123. Epub 2019 Feb 21. PMID: 30826563

Najim O, Seghers S, Sergoyne L, Van Gaver H, Papadimitriou K, Wouters K, Trinh XB, Huizing MT, Tjalma W.(2019).The association between type of endocrine therapy and development of estrogen receptor-1 mutation(s) in patients with hormone-sensitive advanced breast cancer: A systematic review and meta-analysis of randomized and non-randomized trials. 1872(2):188315.
doi: 10.1016/j.bbcancer.2019.188315. PMID: 31647985

Normanno N, Di Maio M, De Maio E, De Luca A, de Matteis A, Giordano A, Perrone F; NCI-Naples Breast Cancer Group.(2005). Mechanisms of endocrine resistance and novel therapeutic strategies in breast cancer. 12(4):721-47.
doi: 10.1677/erc.1.00857. PMID: 16322319

Pedram Razavi et al.,(2018). The Genomic Landscape of Endocrine-Resistant Advanced Breast Cancers. *Cancer Cell*. 2018 Sep 10;34(3):427-438.e6.
doi: 10.1016/j.ccell.2018.08.008. PMID: 30205045

Ribli, D., Horváth, A., Unger, Z., Pollner, P., & Csabai, I. (2018). Detecting and classifying lesions in mammograms with Deep Learning. *Scientific Reports*, 8(1), 4165.

Samala, R. K., Chan, H. P., Hadjiiski, L. M., Helvie, M. A., & Richter, C. D. (2016). Breast cancer diagnosis in digital breast tomosynthesis: effects of training sample size on multi-stage transfer learning using deep neural nets. *Medical Physics*, 43(11), 5333-5344.

Sammons S, Shastry M, Dent S, Anders C, Hamilton E. (2020). Practical Treatment Strategies and Future Directions After Progression While Receiving CDK4/6 Inhibition and Endocrine Therapy in Advanced HR⁺/HER2⁻ Breast Cancer. 20(1):1-11. doi: 10.1016/j.clbc.2019.06.017. PMID: 31780379

Sokol ES et.al.,(2019). Loss of function of NF1 is a mechanism of acquired resistance to endocrine therapy in lobular breast cancer. *Ann Oncol*. 2019 Jan 1;30(1):115-123. doi: 10.1093/annonc/mdy497. PMID: 30423024

Topol, E. J. (2019). High-performance medicine: the convergence of human and artificial intelligence. *Nature Medicine*, 25(1), 44-56.

Wardell SE, Ellis MJ, Alley HM, Eisele K, VanArsdale T, Dann SG, Arndt KT, Primeau T, Griffin E, Shao J, Crowder R, Lai JP, Norris JD, McDonnell DP, Li S. (2015). Efficacy of SERD/SERM Hybrid-CDK4/6 Inhibitor Combinations in Models of Endocrine Therapy-Resistant Breast Cancer. 21(22):5121-5130. doi: 10.1158/1078-0432.CCR-15-0360. PMID: 25991817

Zundeleovich A, Dadiani M, Kahana-Edwin S, Itay A, Sella T, Gadot M, Cesarkas K, Farage-Barhom S, Saar EG, Eyal E, Kol N, Pavlovski A, Balint-Lahat N, Dick-Necula D, Barshack I, Kaufman B, Gal-Yam EN. (2020) ESR1 mutations are frequent in newly diagnosed metastatic and loco-regional recurrence of endocrine-treated breast cancer and carry worse prognosis. *Breast Cancer Res*. 2020 Feb 3;22(1):16. doi: 10.1186/s13058-020-1246-5. PMID: 32014063

Yala, A., Lehman, C., Schuster, T., Portnoi, T., Barzilay, R., & Salama, L. (2019). A deep learning mammography-based model for improved breast cancer risk prediction. *Radiology*, 292(1), 60-66.

Yamashita H, Nishio M, Kobayashi S, Ando Y, Sugiura H, Zhang Z, Hamaguchi M, Mita K, Fujii Y, Iwase H. (2015). Phosphorylation of estrogen receptor alpha serine 167 is predictive of response to endocrine therapy and increases postrelapse survival in metastatic breast cancer. 7(5):R753-64. doi: 10.1186/bcr1285. Epub 2005 Jul 27.PMID: 16168121

Zhang J, Wang Q, Wang Q, Cao J, Sun J, Zhu Z. (2020). Mechanisms of resistance to estrogen receptor modulators in ER+/HER2- advanced breast cancer. 559-572. doi: 10.1007/s00018-019-03281-4. PMID: 31471681

Zhu, J., Zhang, J., & Dong, W. (2019). Deep learning in breast cancer risk assessment: Evaluation of convolutional neural networks on a clinical dataset of full-field digital mammograms. *Journal of Medical Imaging*, 6(3), 031405.

Zhu, W., Huang, Y., Zeng, N., & Wang, M. (2018). Deep learning in medical ultrasound analysis: a review. *Engineering*, 4(5), 730-742.