Breast Cancer detection using Jax Based machine learning models

MSc Research Project
Data Analytics

Nikhil Kadam
Student ID: 22110712

School of Computing
National College of Ireland

Supervisor: Aaloka Anant

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Nikhil Kadam
Student ID:	x22110712
Programme:	Data Analytics
Year:	2023
Module:	MSc Research Project
Supervisor:	Aaloka Anant
Submission Due Date:	14/12/2023
Project Title:	Breast Cancer detection using Jax Based machine learning models
Word Count:	8116
Page Count:	23

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Nikhil Kadam
Date:	30th January 2024

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Breast Cancer detection using Jax Based machine learning models

Nikhil Kadam
x22110712

Abstract

This study investigates the application of JAX-based machine learning models for early breast cancer detection, utilizing the Breast Cancer Wisconsin (Diagnostic) dataset and a Breast Ultrasound Image dataset. The research evaluates a range of models including JAX-based Logistic Regression and Neural Networks, alongside traditional Logistic Regression and Neural Network using Scikit Learn, and deep learning architectures such as CNN, InceptionNet, DenseNet121, and VGG19.

Key findings reveal that the Scikit Learn-based Logistic Regression model excels in the structured Wisconsin dataset, achieving approximately 99% accuracy, while the Neural Network model also shows high efficiency. JAX-based models, however, exhibit mixed results, with the Neural Network particularly underperforming, indicating challenges in complex model implementations within JAX. In the image dataset evaluation, VGG19 outperforms others with about 77% accuracy, highlighting its strength in image-based data analysis. In contrast, CNN and DenseNet121 show moderate effectiveness, and InceptionNet falls behind, suggesting a potential mismatch with the dataset's features.

1 Introduction

Breast cancer, increasingly prevalent in today's society, has emerged as a significant health challenge across the globe Yadav et al. (2023). This rise necessitates a paradigm shift in detection and diagnostic methodologies. In this evolving landscape, machine learning (ML) emerges as a guiding light, offering innovative approaches to early detection and effective treatment strategies that can dramatically influence patient outcomes and reshape healthcare practices (Desai and Shah; 2021)..

JAX is a cutting-edge open-source numerical computing library by Google, which is a game-changer in the field of machine learning Lange (2023). JAX is known for its ability to execute complex mathematical operations, which is a cornerstone for deep learning models. By integrating JAX in breast cancer detection, we can expect a new era in medical diagnostics, promising accuracy and speed in handling intricate datasets, including medical imaging and patient histories (Desai and Shah; 2021).

The motivation behind this study is the urgent need to revolutionise breast cancer detection methods. Conventional techniques, while beneficial, are marred by limitations such as cost, invasiveness, and potential inaccuracies Yue et al. (2018). JAX-based machine learning models hold the potential to transcend these barriers, presenting a non-invasive, cost-effective, and remarkably accurate diagnostic alternative. This innovation is particularly vital in regions where advanced medical facilities are scarce Afaya et al. (2022) .

The significance of this endeavour extends across several dimensions. It adds a substantial chapter to the corpus of medical machine learning, especially in deploying JAX for health data analysis. More importantly, it tackles a global health menace - breast cancer. Early detection, which this study facilitates, is pivotal in breast cancer treatment, significantly uplifting survival rates Yadav et al. (2023). Additionally, democratising access to advanced diagnostic tools through JAX-based ML models could revolutionise healthcare in diverse settings Afaya et al. (2022).

This research's cornerstone is its innovative use of JAX-based machine learning algorithms for breast cancer detection. By harnessing JAX's power in processing complex data, the study sheds new light on cancer diagnostics. It validates the effectiveness of JAX in medical diagnostics and sets the stage for its expansive application in health-care. Moreover, it offers crucial insights into integrating sophisticated computing technologies with medical research, potentially catalysing future studies and technological breakthroughs in healthcare.

The upcoming chapter deals with the critical assessment of the current state of the art in breast cancer diagnosis. The following chapters deal with the design of the system, its implementation and evaluation.

1.1 Background

In recent years the most frequent cancer among women is breast cancer. In this case, the early detection of the disease can increase the chances of survival for the patient. Due to the efficiency and speed at which the JAX neural network is produced while coupled with Numpy and Python, it is widely adopted by machine learners. This neural network helps to save memory energy and cost. Using AI-based technology has become one of the most important topics in the healthcare sector. In the case of cancer, time result in it is the most important factor in science late detection can lead to late treatment for which the patient can suffer more and it also can risk their life. Sometimes many doctors are insecure about the diagnosis test result since the lives of the patients depend on it. Hence the second affirmation that the AI generates helps the doctors to act faster. The AI models can assist doctors in making effective decisions on time. The performance of cancer detection depends on the hardware that is used in the AI technology. In this case, the Jax neural network embedded devices can be very effective in detecting health issues

1.2 Concept

Improving health care outcomes of patients requires the early diagnosis of breast cancer, and JAX or "Just Another extension" integration with neural networks and logistic regression offers an effective method to boost cancer diagnosis precision. Breast cancers are the most common type of cancer among women Yue et al. (2018). The neural net-work JAX can provide results quickly, and the numerical computing library JAX is very effective for various machine learning applications. The auto guard features of JAX speed up the utilisation process in logistic regression, a key method for binary classification problems, by automatically computing gradients. This makes it attainable for the model to modify its parameters, enhancing its capacity to identify trends in medical data that help to diagnose early breast cancer. The logistic regression model can handle big datasets that are repeatedly contended in medical research because it benefits from quicker computations when implemented using JavaScript.

The scalability of the model is improved by JAX capabilities, which allow it to handle a mixture of characteristics linked to risk elements for breast cancer. By using JavaScript, logistic regression achieves more power as an early detection tool since it can better forecast outcomes and efficiently determine complex relationships in the data. Beyond logistic regression, JAX is important for neural network optimisation in the early identification of breast cancer. Neural networks deliver a more refined method of analysing medical data because of their ability to recognise complex patterns. Neural networks are empowered by JAX’s autographed components, which compute gradients automatically during training and make it more comfortable to examine high-dimensional parameter spaces.

When JAX is used in neural networks, training deep architectures becomes more efficient, leading to the development of more evolved models that can recognise minor details that point to early-stage breast cancer. The neural network can reveal complex linkages in the data and provide a more refined knowledge of breast cancer risk factors because of its capacity to develop hierarchical representations. In the medical setting, this is particularly important because quick discovery can have a big impact on patient outcomes. logistic regression with JAX abilities merged with neural networks is a powerful tool for early breast cancer identification. Integration of JAX allows these models to efficiently explore large and complicated medical datasets. This method can enhance patient prediction and treatment results by increasing the field of medical diagnostics and assisting in the early and more precise detection of breast cancer.

2 Related Work

The research by Yadav et al. (2023), elaborates that breast cancer is a disease in which the breast cells develop uncomfortably and unnaturally which creates a mass called a tumour. If this mass is not addressed effectively then it can spread to the other biddy parts including the lungs, liver and bones. Both men and women are affected by this disease however men are at more risk than women. The study further showcases that in the last three years, almost 7.8 million women have been diagnosed with breast cancer. The World Health Organization has announced breast cancer as the most frequent cancer worldwide by the end of 2020. This research investigates breast cancer detection while applying machine learning algorithms.

The datasets of this research include breast cancer detection databases from mammography imaging” as well as “Wisconsin” datasets. Different cancer detection models have been established to find the most effective one. Machine learning and deep learning algorithms have been implemented to find and predict breast cancer. These deep learning and machine learning algorithms have used the performance of each classifier to find the most suitable outcome. It has been observed while reviewing the study that in ML algorithm application a proper dataset is required tin enhance the accuracy.

The research which is conducted by Sardouk et al. (2019), analyses how effectively data mining can classify breast cancer. The importance of regular investigation and proper requisitions has also been established in the study. The data mining process can effectively BMI, age and sugar routine databases. detect breast cancer while using some parameters like- age, BMI and sugar level of the patient. The study highlights regular testing and preventative actions to emphasise the significance of early diagnosis of breast cancer. Data science, and especially data mining, plays a critical role in making it easier to find patterns in large datasets, such as the” Coimbra” dataset with ten predictors.

The study uses six methods, including Artificial Neural Networks (ANNs), and com-pares how well they function in MATLAB and WEKA. The” Knowledge Discovery in Databases” (KDD) approach is essential for tackling classification and clustering problems in large databases. The study examines several critical factors, clustering performance indicators, such as hierarchical clustering, and the scalability of data mining methods, that are necessary for managing the increasing volume of breast cancer data. The study also analyses the consequences of clustering performance indicators, hierarchical clustering, and data mining method scalability—all of which are essential for handling the growing amount of cancer data.

According to (Desai and Shah; 2021), the investigation of artificial neural networks, notably Multi-Layer Perceptron Neural Network (MLP) and Convolutional Neural Net-work (CNN), concerning early breast cancer diagnosis is thoroughly examined in this paper. The paper compares MLP and CNN in detail, highlighting the differences in their designs and functionalities. The accuracy of breast cancer detection, which is essential for early cancer diagnosis, is the main area of focus. Interestingly, the article discovers that CNN outperforms MLP in accuracy by a small margin. Still, it effectively emphasises how important it is to conduct additional research with both networks using the same datasets and settings. This is in line with our main research title,” JAX Logistic Regression and JAX Neural Network in the Context of Early Breast Cancer Detection,” since it highlights the importance of comparing and analysing various approaches in a nuanced manner to improve breast cancer diagnosis accuracy.

The findings of (Shubham and Kamalraj; 2022), highlight how breast cancer poses a significant global danger to women’s health and highlight the revolutionary potential of artificial intelligence (AI) in identifying whether a tumour is benign or malignant. The paper makes a strong case for the superiority of AI techniques over conventional diagnostic accuracy, notably K-Nearest Neighbours (KNN), Support Vector Machine (SVM), and Decision Tree Classifier (DT). Surprisingly, the SVM classifier wins hands down, producing better and more accurate results, especially when trained on large datasets. This is in perfect harmony with the title of our main study,” JAX Logistic Regression and JAX Neural Network in the Context of Early Breast Cancer Detection,” which emphasises how important it is to investigate cutting-edge AI methods for improved precision in early breast cancer diagnosis.

Also, the article by Yue et al. (2018), offers a thorough examination of machine learning (ML) applications in the diagnosis and prognosis of breast cancer, and it fits in perfectly with the main research topic. Acknowledging breast cancer as a widespread worldwide health concern, the study emphasises the critical function of prompt diagnosis in enhancing prognosis and survival rates. Artificial neural networks (ANNs), support vector machines (SVMs), decision trees (DTs), and k-nearest neighbours (k-NNs) are among the machine learning (ML) approaches that are examined and their contributions to forecast modelling and pattern categorization of breast cancer are skilfully discussed. Notably, the results are more credible because the Wisconsin Breast Cancer Database (WBCD) was used as the main source of data. The article’s focus on machine learning’s distinct benefits in identifying important characteristics is in line with our research’s goal of utilising cutting-edge methods to increase the accuracy of early breast cancer detection. The article by Osman (2017), offers a promising development in the diagnosis of breast cancer by combining a Support Vector Machine (SVM) with a Two-Step Clustering Technique; this fits in well with our general research on the use of JAX Neural Network and JAX Logistic Regression for Early Breast Cancer Detection. Through tackling the crucial issues of misclassification and diagnostic accuracy in breast tumour classification, the suggested hybrid approach shows an impressive 99.1% accuracy on the UCI-WBC dataset. Combining SVM and Two-Step Clustering yields significant improvements in accuracy when it comes to identifying hidden patterns of benign and malignant tumours. The hybrid approach is notably better at diagnosing breast cancer than modern classification systems, according to the trial data. This is in line with our study objective, which is to investigate cutting-edge methods to improve diagnostic accuracy. It also highlights the significance of novel strategies, like the Two-Step-SVM technique, in the field of early breast cancer diagnosis.

Additionally, the article Min et al. (2017), investigates a new approach to breast cancer diagnosis using Thermal Infrared Image Analysis, which fits very well with our main study on the use of JAX Neural Network and JAX Logistic Regression in Early Breast Cancer diagnosis. The research, which makes use of the growing use of thermal imaging cameras, promotes a non-invasive, low-cost diagnostic technique that is particularly helpful for expectant mothers and women with particular breast characteristics. Interestingly, the suggested method makes use of thermal imaging, first testing on female Cambodian subjects, and analysis through the use of Shannon entropy and logistic regression. This novel approach addresses radiation exposure concerns and offers a viable substitute for conventional breast cancer screening. This is in line with the goal of our research, which is to investigate various methods while highlighting the significance of secure and reliable diagnostic methods for the early diagnosis of breast cancer.

Moving on, the article by Kumar et al. (2020) offers a Machine Learning-based Optimised Prediction Method, which substantially advances the general research on JAX Logistic Regression and JAX Neural Network in Early Breast Cancer Detection. The study highlights the need for automated detection to save time and prevent disease spread, given that breast cancer is one of the main causes of high mortality rates among women. The paper carefully assesses classification accuracy, sensitivity, specificity, and other factors by comparing four machine learning algorithms: Logistic Regression, SVM, KNN, and Naive Bayes. SVM emerges as the top performer with an astounding accuracy of 98.24% when various hyperparameters are manually assigned to each algorithm. This demonstrates how well-sophisticated techniques support our goal of optimising prediction tools for early breast cancer detection and highlights how important they are. Another article by Omondiagbe et al. (2019), explores Machine Learning Classification Methods for Breast Cancer Diagnosis, which substantially advances our general research on JAX Logistic Regression and JAX Neural Networks in Early Breast Cancer Detection. The report acknowledges that breast cancer is a serious and pervasive health concern and highlights the urgent need for precise computer-aided detection (CAD) systems. This study uses the Wisconsin Diagnostic Breast Cancer Dataset and focuses on Support Vector Ma-chines, Artificial Neural Networks, and Naïve Bayes. Remarkably, the hybrid approach that combines linear discriminant analysis (LDA) for dimensionality reduction and then applies it to Support Vector Machine achieves impressive results in terms of accuracy (98.82%), sensitivity (98.41%), specificity (99.07%), and area under the receiver operating characteristic curve (0.9994). This demonstrates how hybrid strategies can improve diagnostic precision in the early diagnosis of breast cancer, which is in perfect alignment with our research goal of investigating innovative methodologies.

The article by Shawarib et al. (2020), makes a substantial addition to the prediction of survival and prognosis for breast cancer by using an Artificial Neural Network (ANN) model in the context of a JNN. The suggested model performs well on the Wisconsin Breast Cancer dataset, addressing the ongoing issues with early prognosis and survival prediction rates, and attaining an astounding accuracy of 99.57%. The correctness of the model is further confirmed by testing it on Haberman's Breast Cancer Survival data-set, which yielded an accuracy of 88.24%. Interestingly, these results outperform the effectiveness of earlier supervised learning techniques. The authenticity of the results is increased by the study's focus on real-world datasets without preprocessing. The use of terms like" prediction,"" JNN,"" ANN," and" breast cancer" highlights the novel and successful nature of the suggested strategy in improving breast cancer diagnostic and survival prediction, and it appropriately represents the study's emphasis.

The article by Kumar et al. (2017), explores the" Prediction of Breast Cancer using Voting Classifier Technique," which adds significantly to our overall research on JAX Logistic Regression and JAX Neural Network in Early Breast Cancer Detection. Recognising breast cancer as a serious health concern, the study highlights how important it is to get a diagnosis as soon as possible. It makes use of an ensemble approach for better classification by comparing and integrating various supervised learning classification algorithms through the voting classifier technique. The results have more confidence because the Wisconsin University database was used. The article is in perfect alignment with our research goal of investigating various approaches and demonstrating the capacity of ensemble techniques to improve classification accuracy in early breast cancer prediction. This emphasises how important it is to combine different models for a more reliable and accurate diagnosis procedure in keeping with our main area of study.

In addition to this, the article by Hamed et al. (2021), makes a substantial contribution to our overall study of the role of JAX Neural Network and JAX Logistic Regression in Early Breast Cancer Detection. It presents a system of computer-aided design (CAD) that uses several machine-learning techniques to detect breast cancer. On the BC Wisconsin dataset, random forest shows up as the best performer, achieving an astounding 99% accuracy and F-measure. This demonstrates the promise of machine learning algorithms, particularly random forest, in boosting accuracy for early breast cancer detection, and fits in perfectly with our research goal of optimising predictive methodologies.

On the other hand, an article by Singh and Singh (Singh and Singh; 2020), focuses on the integration of Infrared Breast Thermography with artificial neural networks, which greatly enhances our general study on JAX Logistic Regression and JAX Neural Network in Early Breast Cancer Detection. Recognising the frequency of breast cancer, it emphasises the value of precise early detection and presents thermography as a non-invasive supplement to mammography. The paper's focus on computer-aided detection systems that use machine learning to classify breast thermograms is in line with our study objective of investigating cutting-edge methods. Interestingly, a practical approach is shown by using numerical simulation to address false positives. The article's suggestions for the future are in perfect harmony with the goal of our research, which is to use the latest developments in machine learning to improve early breast cancer detection in real-time.

Another article by Mangukiya et al. (2022), aims to improve breast cancer diagnosis by utilising the Wisconsin Breast Cancer Dataset to conduct a thorough investigation of machine learning techniques. In light of the study's recognition of the worldwide effect of breast cancer, early detection is crucial for increased survival rates. The study carefully assesses accuracy, precision, sensitivity, and specificity using a Support Vector Machine, Decision Tree, Naive Bayes, K K-nearest neighbours, Adaboost, XGboost, and Random Forest. With the lowest mistake rate and a high accuracy of 98.24%, XGboost stands out as the most efficient method. The focus on various methods for timely and precise identification is in line with the main idea of Breast Cancer Detection using Machine Learning.

3 Research Methodology

Data mining research can be conducted through Knowledge Discovery in Databases (KDD) and Cross-Industry Standard Process in Data Mining (CRISP-DM). These methodologies differ in practical applications. CRISP-DM is a process-based methodology, whereas the KDD is more research-oriented. To conduct the presented research, the KDD methodology has been selected as the deployment phase of the CRISP-DM is not necessary in the present context.

Based on the KDD methodology Azevedo and Santos (2008), this study comprises 5 phases viz. Data Selection, Data Preparation, Data Transformation, Data Mining and Evaluation.

The methodology is also divided into experiments. In the first experiment a structured dataset is used whereas a non-structured image dataset is used in the second experiment.

3.1 Experiment 1: Breast Cancer Wisconsin Dataset

Figure 1 below shows the methodology of Breast Cancer detection using JAX-based Logistic Regression and JAX-based Neural Network.

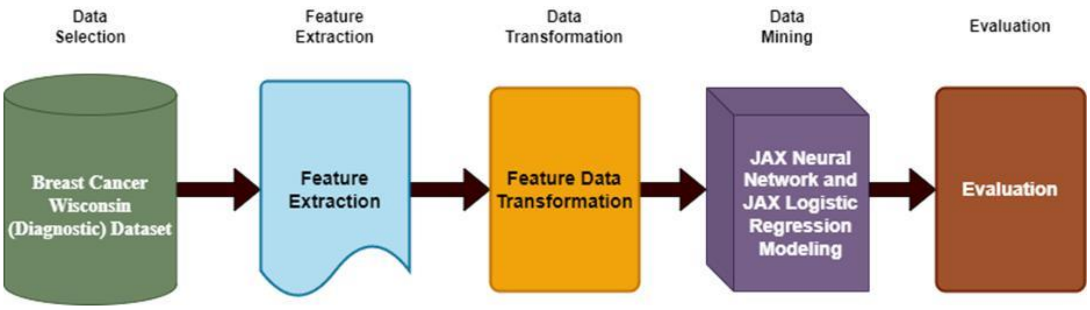

Figure 1: Methodology flow for the Breast Cancer Wisconsin Dataset

3.1.1 Data Selection

The Breast Cancer Wisconsin (Diagnostic) dataset, housed at the UCI Machine Learning Repository, is a comprehensive dataset for diagnosing breast cancer. It contains 569 instances with 30 real-valued features calculated from digitised images of fine needle aspirates of breast masses. With a focus on the characteristics of the cell nuclei present in the image. The dataset includes shape-related features of the cell nuclei such as radius, texture, perimeter, area, smoothness, compactness, concavity, concave points, symmetry, and fractal dimension. Additionally, it includes an ID number and a diagnosis label (M = malignant, B = benign). Wolberg and Street (1995)

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean	concave points_mean
0	842302	M	17.99	10.38	122.80	1001.0	0.11840	0.27760	0.30010	0.14710
1	842517	M	20.57	17.77	132.90	1326.0	0.08474	0.07864	0.08690	0.07017
2	84300903	M	19.69	21.25	130.00	1203.0	0.10960	0.15990	0.19740	0.12790
3	84348301	M	11.42	20.38	77.58	386.1	0.14250	0.28390	0.24140	0.10520
4	84358402	M	20.29	14.34	135.10	1297.0	0.10030	0.13280	0.19800	0.10430
5	843786	M	12.45	15.70	82.57	477.1	0.12780	0.17000	0.15780	0.08089
6	844359	M	18.25	19.98	119.60	1040.0	0.09463	0.10900	0.11270	0.07400
7	84458202	M	13.71	20.83	90.20	577.9	0.11890	0.16450	0.09366	0.05985
8	844981	M	13.00	21.82	87.50	519.8	0.12730	0.19320	0.18590	0.09353
9	84501001	M	12.46	24.04	83.97	475.9	0.11860	0.23960	0.22730	0.08543

Figure 2: shows the samples from the dataset.

The dataset consists of 31 features, out of which a feature named Unnamed:32 is an empty column. This can be seen in Figure 3. The figure also shows the data types of the features present in the dataset. It shows that all the features except for the ‘diagnosis’ feature are numerical. The diagnosis feature is the dependent variable in the dataset and take string values (Figure 2).

```
 #   Column                   Non-Null Count  Dtype
---  ------                   --------------  -----
 0   id                       569 non-null    int64
 1   diagnosis                569 non-null    object
 2   radius_mean              569 non-null    float64
 3   texture_mean             569 non-null    float64
 4   perimeter_mean           569 non-null    float64
 5   area_mean                569 non-null    float64
 6   smoothness_mean          569 non-null    float64
 7   compactness_mean         569 non-null    float64
 8   concavity_mean           569 non-null    float64
 9   concave points_mean      569 non-null    float64
 10  symmetry_mean            569 non-null    float64
 11  fractal_dimension_mean   569 non-null    float64
 12  radius_se                569 non-null    float64
 13  texture_se               569 non-null    float64
 14  perimeter_se             569 non-null    float64
 15  area_se                  569 non-null    float64
 16  smoothness_se            569 non-null    float64
 17  compactness_se           569 non-null    float64
 18  concavity_se             569 non-null    float64
 19  concave points_se        569 non-null    float64
 20  symmetry_se              569 non-null    float64
 21  fractal_dimension_se     569 non-null    float64
 22  radius_worst             569 non-null    float64
 23  texture_worst            569 non-null    float64
 24  perimeter_worst          569 non-null    float64
 25  area_worst               569 non-null    float64
 26  smoothness_worst         569 non-null    float64
 27  compactness_worst        569 non-null    float64
 28  concavity_worst          569 non-null    float64
 29  concave points_worst     569 non-null    float64
 30  symmetry_worst           569 non-null    float64
 31  fractal_dimension_worst  569 non-null    float64
 32  Unnamed: 32              0 non-null      float64
```

Figure 3: Dataset information

The distribution of the ‘diagnosis’ column values is shown in Figure 4 below. It is evident from the feature that the number of samples belonging to the Malignant (M) class are less compared to that of the Benign (B) class.

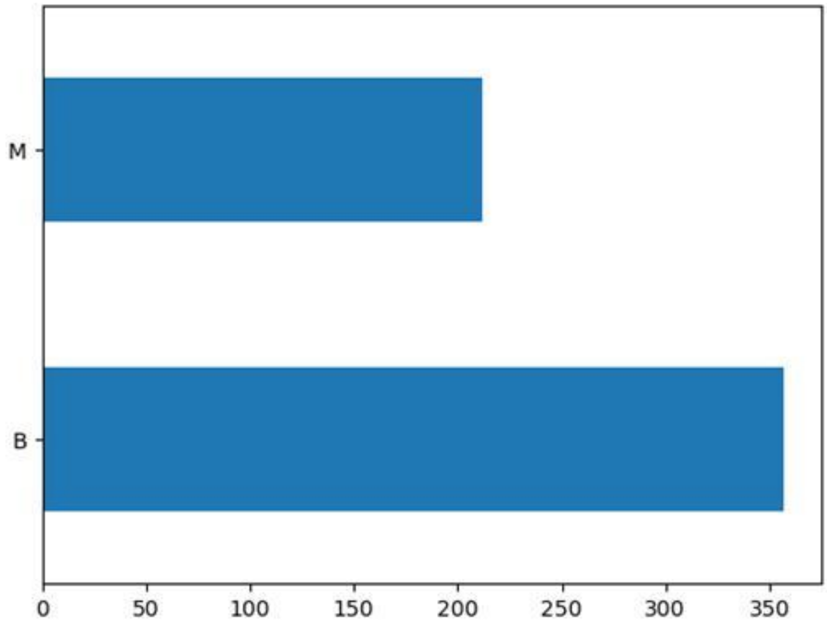

Figure 4: Distribution of class

Once the absence of the noisy data is observed, a correlation heatmap depicting the correlation between the features is observed, helping to study the features better.
The other dataset used in the study is discussed below.

3.1.2 Feature Selection

The feature selection in the study is done using XGBoost classifier implementation. Feature importance can be calculated using tree-based classifiers such as XGBoost. The importance of the feature in the study is calculated using hyperparameter optimization using the GridSearchCV algorithm. The GridSearchCV uses numerous combinations of hyperparameter values and cross-validation to obtain the most accurate classification model. In this study, the model is cross-validated with 3 cross-validation. Table 1 enlists the hyperparameters and their set of values selected for tuning.

Table 1: Hyperparameter tuning for feature selection

Hyperparameter	Value
max_depth	1,2,3,4,5,7
learning _rate	0.5, 0.1, 0.01, 0.05
gamma	0, 0.25, 1, 3, 5, 7
reg_ lambda	0, 1, 10
scale_pos_weight	1,3,5
subsample	0.8
colsample_bytree	0.5

Once, the most importance features from the dataset are obtained. A smaller dataset is created which then can be transformed to suit the modelling task at hand.

3.1.3 Data Transformation

For the presented dataset, the following data transformation steps are undertaken

Label Encoding Label encoding is a process of converting non-numerical features in the dataset into numerical form. This helps the machine learning models to work on them as most of the machine learning models require the data to be in numerical form. As seen in the dataset, the diagnosis column is the only column in the dataset that is non-numerical. Hence, it is necessary to convert the mentioned column into numerical form. The study's label encoding step is performed utilizing the Scikit Learn library's LabelEncoder function. This function first finds the unique values in the dataset in the respective column, which are then arranged in alphabetical order, meaning the value B in the diagnosis column precedes the value M. After this arrangement, numbers from 0 are given to the non-numerical values in ascending order. Hence, the value B becomes 0 and the value M becomes 1 converting the classes into numerical form.

Class balancing with Synthetic Minority Oversampling

(SMOTE) Class balancing is necessary in data mining applications where there is a difference between the count of samples belonging to the classes in the dataset. This is because, the machine learning models tend to favour the class in majority during training hampering the reliability of the model. To avoid this, performing class balancing becomes inevitable. As seen from figure 8 below. The number of samples belonging to class 1 is less compared to that of class 0.

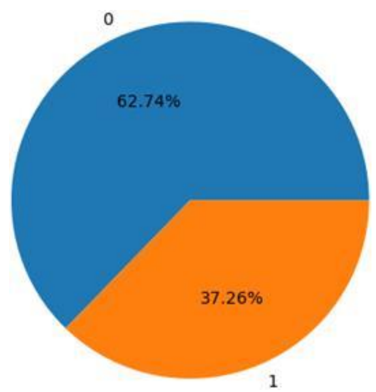

Figure 5: Class balance in the dataset

The class balancing in the study is performed using the Synthetic Minority Over-sampling Technique (SMOTE). In this technique additional samples belonging to the class in minority, i.e. class M (1) in this case, are generated by picking a random sample from the underrepresented class. It then identifies a small number of nearest neighbours for this sample in the feature space. To create new, synthetic samples, SMOTE selects one of these neighbours at random, draws a line between the chosen example and this neighbour, and picks a random point along this line. This point, obtained by multiplying the connecting vector by a random number between 0 and 1, becomes the new sample. This method of linear interpolation is repeated until the dataset achieves a more balanced class distribution, effectively enriching the minority class with new, varied examples.

Standard Scaling (Standardisation) Many machine learning algorithms especially that use gradient descent optimization (like neural networks), perform better when the data is standardized. When features are on the same scale, it helps the algorithm to converge faster. Also, without standardization, features with higher magnitudes can dominate the model learning, leading to a bias towards these features. Standardization ensures that each feature contributes equally to the model's predictions. Figure 6 below shows the statistical description of the dataset features.

	id	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean	concave points_mean
count	5.690000e+02	569.000000	569.000000	569.000000	569.000000	569.000000	569.000000	569.000000	569.000000
mean	3.037183e+07	14.127292	19.289649	91.969033	654.889104	0.096360	0.104341	0.088799	0.048919
std	1.250206e+08	3.524049	4.301036	24.298981	351.914129	0.014064	0.052813	0.079720	0.038803
min	8.670000e+03	6.981000	9.710000	43.790000	143.500000	0.052630	0.019380	0.000000	0.000000
25%	8.692180e+05	11.700000	16.170000	75.170000	420.300000	0.086370	0.064920	0.029560	0.020310
50%	9.060240e+05	13.370000	18.840000	86.240000	551.100000	0.095870	0.092630	0.061540	0.033500
75%	8.813129e+06	15.780000	21.800000	104.100000	782.700000	0.105300	0.130400	0.130700	0.074000
max	9.113205e+08	28.110000	39.280000	188.500000	2501.000000	0.163400	0.345400	0.426800	0.201200

Figure 6: Statistical Description of the Dataset

From the statistical description, it can be observed that the values for the features seem to change a lot, e.g. the mean value for the concave points mean feature is 0.048919, whereas the mean value for the area mean feature is as high as 654.8891. It is visible that there is a huge different between mean values of these features. The standardization in this study is performed on the dataset features through the pipelining option in which multiple columns are standardized in one go.

Data Splitting Once the data is standardized, it is split into two parts: training set and testing test. The split in the dataset is at a ratio of 70:30. The 70% of the data is considered for training whereas the remaining 30% is used for testing the models.

3.1.4 Modelling

Neural Network Neural networks are computer models based on how the human brain is structured and works. They consist of layers of nodes, or" neurons," that can perform simple calculations and are interconnected. These neurons are arranged in layers: an input layer, hidden layers, and an output layer (see Figure 7). The input layer receives information, hidden layers work on it, and the output layer produces the result. Neural networks excel at identifying patterns and learning from data, which makes them useful for many applications such as speech and image recognition, natural language processing, and prediction. They learn by adjusting the connection weights (denoted as Wi) based on the degree of error in their predictions, a process known as training. Neural networks are a crucial component of modern AI systems because they can solve a wide range of complex, non-linear problems efficiently.

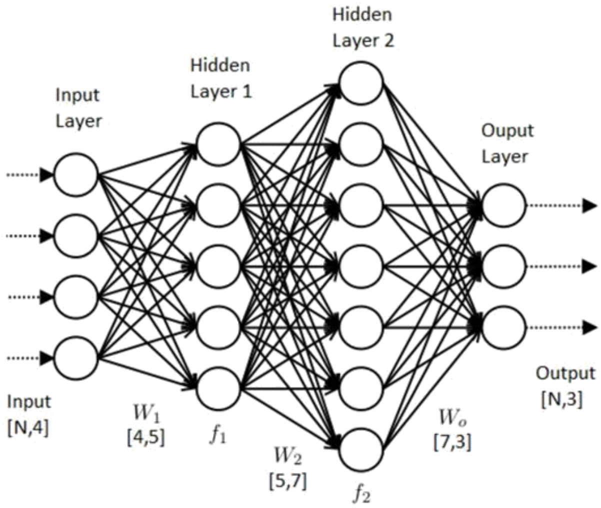

Figure 7: Structure of a typical neural network

Logistic Regression Logistic regression is a statistical method used to solve problems with two possible outcomes. It estimates the probability of an event or a class occurrence by fitting data to a logistic curve.

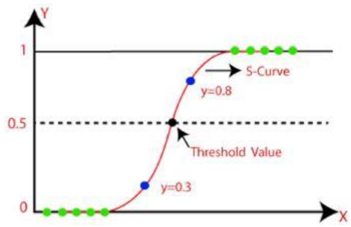

Figure 8: Fitting the data to a logistic curve

The output of logistic regression is a value between 0 and 1 that indicates the likelihood of a specific type of event happening. This method is advantageous when there are only two possible results, like identifying spam emails. Logistic regression is easy to use and comprehend, and it works well with linear relationships, making it a popular choice in various fields, including medicine, economics, and social sciences.

3.1.5 Evaluation

The models implemented in this experiment are evaluated based on four evaluation metrics: Accuracy, Precision, Recall, and F1-score.

3.2 Experiment 2: Breast Ultrasound Image Dataset

3.2.1 Data Collection

The Breast Ultrasound Images dataset comprises ultrasound scans of the breasts of women aged 25 to 75. It includes 600 female participants, resulting in a total of 780 scans. Each scan, averaging 500x500 pixels in size, is saved in PNG format. Accompanying each scan is its corresponding 'ground truth' image. These scans are further classified into three categories: normal, benign, and malignant. The dataset for the study is obtained from the Kaggle repository [1]

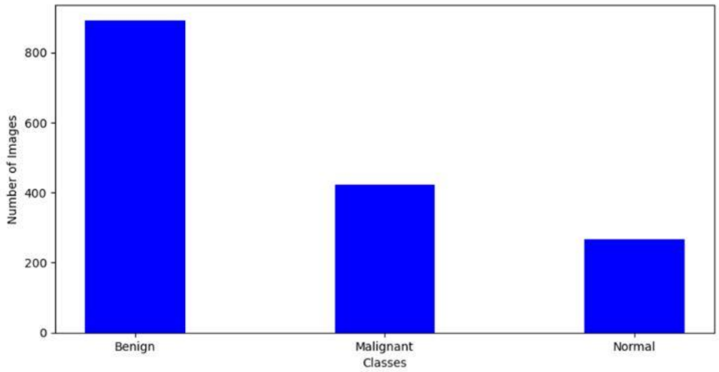

Figure 9: Class distribution in image dataset

3.2.2 Image Augmentation

As can be seen from Figure 9, the classes in the dataset are not balanced and thus require balancing. The balancing in the image dataset is achieved through a process known as image augmentation in which additional samples are generated from the images present in the dataset. The additional samples can be generated through a range of operations that can be performed on an image.

Table 2 below enlists the operations that have been performed in the study for augmenting the images in the dataset. The image augmentation in the study is performed using the ImageDataGenerator module of the Keras library. A generator object is created to augment images using the parameters given in Table 2. The object divides the images into training and testing sets while augmentation. The splitting ratio of the dataset is 70:30.

Table 2: Augmentation parameters and their values

Augmentation Parameter	Purpose	Parameter Value
Rescale	Normalizes the image	1/255
Rotation _ range	Rotates the image	40
Width _shift_ range	Changes the width keeping overall size same	0.2
Zoom _ range	Zooms into the image	0.2
Height _ shift_ range	Changes the height keeping size same	0.2
Shear_ range	Elongates the image contents in a direction	0.2
Horizontal _ flip	Flips the image horizontally	True
Fill _ mode	Fills the extrapolated part	nearest

3.2.3 Modelling

Once the divided training and test data is available, the following deep learning models are trained on the images.

[1] https://www.kaggle.com/datasets/aryashah2k/breast-ultrasound-images-dataset

Convolutional Neural Network a Convolutional Neural Network (CNN) is a powerful deep learning algorithm that has transformed the field of computer vision. Essentially, a CNN is trained to recognize patterns in images, which makes it incredibly effective at tasks such as object detection and image classification. Unlike traditional algorithms, CNNs automatically and adaptively learn spatial hierarchies of features from input im-ages. This is accomplished through a series of layers, each of which extracts different features from the input. The early layers capture simple features, such as edges and textures, while the deeper layers can recognize more complex features, such as shapes and objects.

The typical architecture of a CNN includes convolutional layers, pooling layers, and fully connected layers. Convolutional layers employ a set of learnable filters that help the network identify various features in the input. Pooling layers, which usually follow the convolutional layers, reduce the spatial size of the representation. This reduces the computational power required and helps to prevent overfitting. Finally, fully connected layers, which resemble traditional neural network layers, use the features extracted by the convolutional and pooling layers to classify the input image into various categories. This streamlined and efficient processing makes CNNs highly effective for image-related tasks.

Inception Net Inception Net, known for its unique architecture, redefined efficiency in Convolutional Neural Networks (CNNs). It’s built on the idea of a ’network within a network’ and employs parallel convolutions of different sizes to capture information at various scales. The key feature is its inception modules, which allow the network to choose from different kernel sizes (1x1, 3x3, 5x5) and a max pooling operation simultaneously. This design reduces computational cost while increasing the network’s depth and width. Inception Net has shown remarkable performance in image classification tasks, particularly in Google’s ImageNet competition.

DenseNet121 DenseNet121, a member of the Densely Connected Convolutional Net-works (DenseNet) family, stands out for its unique approach to connectivity. In DenseNet, each layer is connected to every other layer in a feed-forward fashion. For each layer, the feature-maps of all preceding layers are used as inputs, and its own feature-maps are used as inputs into all subsequent layers. This creates very dense connections, hence the name DenseNet. This architecture strengthens feature propagation, encourages feature reuse, and substantially reduces the number of parameters, making it both efficient and powerful in capturing complex features in images.

VGG19 VGG19, part of the VGG (Visual Geometry Group) series, is celebrated for its simplicity and depth. It consists of 19 layers, including 16 convolutional layers, and it follows an architecture of increasingly deep convolutional networks. VGG19 uses small 3x3 convolutional filters throughout, which allows it to capture fine details from the input images. Despite its relative simplicity, VGG19 has shown impressive performance in various image recognition tasks. Its depth and use of small convolutional filters enable it to learn a wide array of complex and subtle features, although this depth can make it computationally intensive to train and deploy.

3.2.4 Evaluation

The models implemented in this experiment are evaluated based on the accuracy achieved by the models in classifying the Breast Ultrasound Images.

4 Design Specification

This chapter delves into the detail about the architecture of the system employed in the study. The system architecture consists of two layers which are presentation unit and processing unit. The presentation unit is a user side unit that is responsible for visualising the datasets as well as the evaluation reports of the models. The processing unit on the other hand handles various tasks ranging from reading the dataset, various pre-processing steps for both the experiments, the modelling and evaluation parts as well.

Figure 10 below shows the system architecture of the study.

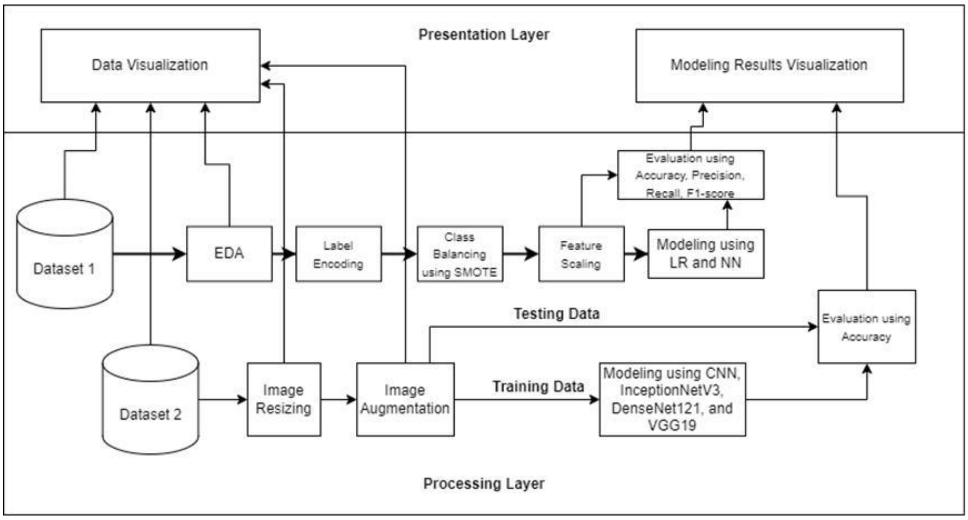

Figure 10: System architecture

5 Implementation

This chapter of the report deals with the implementation of the system presented. The implementation is divided in two experiments wherein experiment 1 machine learning models of Logistic Regression and Artificial Neural Network are implemented using Scikit Learn library and JAX framework on the Breast Cancer Wisconsin Dataset. In experiment 2, deep learning architectures such as CNN, Inception Net, DenseNet, and VGG19 used for modelling the Breast Ultrasound Image Dataset through transfer learning using Keras library for Python.

5.1 Experiment 1

5.1.1 Implementation of Logistic Regression using Scikit Learn

The logistic regression (LR) model is implemented in the study using Scikit Learn library's LogisticRegression object of the linear model module of the library. Table below lists the hyperparameter chosen for the model.

Table 3: Hyperparameters for Logistic Regression model

Hyperparameter	Value
Max iter	100

The model is first instantiated using the function and then fit onto the training set of the data. Once trained the model is used to predict the classes for the test data using the predict () function of the model object. The hyperparameters for the model are chosen through random search and are chosen when the highest accuracy for the model is achieved.

5.1.2 Implementation of Neural Network using Scikit Learn

Similar to the LR, the neural network is implemented using the Scikit Learn library's neural network module through the MLPClassifier object. The hyperparameters for the model are chosen through a random search similar to that of the LR model. The following table depicts the hyperparameters chosen for the model. The MLPClassifier is built with 1 hidden layer with 128 neurons.

Table 4: Hyperparameters for the Neural Network model.

Hyperparameter	Value
Activation	tanh
Max iter	1000

The model is then trained on the training set and the trained model is then used to predict the classes for the testing set using predict () function of the model object.

5.1.3 JAX-based Logistic Regression

JAX is a library for fast numerical computing that was made for research that uses machine learning. It adds new features to the NumPy and SciPy libraries for auto grad and XLA (Accelerated Linear Algebra) compilations. To put it simply, JAX is based on a foundation that makes it easy and quick to use both machine learning and scientific computing algorithms.

The logistic regression through JAX is implemented using the mathematical equations that define the LR model. These equations are implemented as functions and are given in Figure 11 below.

```
def logistic(r):
    return 1 / (1 + jnp.exp(-r))

def predict(c, w, X):
    return logistic(jnp.dot(X, w) + c)
```

Figure 11: Functions implementing LR equation for definition and prediction

Table 5: Choice of hyperparameters for the JAX LR implementation

Hyperparameter	Value
C_0	1.50
W_0	9.0e-1

C–0 and W-0 are the hyperparameters for the LR function. Here W0 is a vector that multiplies each feature value with 9e-1. A dot product of features (X) and W0 is obtained which is added to C0 to get the value r. This value is then used to obtain the class by getting the value of the logistic function.

5.1.4 JAX-based Neural Network

Mathematically, a neural network can be defined using the equation below.

$$f(x) = \eta\ (W_2 \cdot \eta\ (W_1 \cdot x + b_1) + b_2) \tag{1}$$

W1 and W2 are the weight matrices for the input and output weights for the connections between the input layer, the hidden layer, the hidden layer, and the output layer, respectively.

Table 6: Choice of hyperparameters for the JAX LR implementation

Hyperparameter	Value
n_iter	100
eta	2e-2
tol	5e-6
w	9e-1
c	1.5

In the JAX-based neural network implementation, the process begins with input data being fed into the model. Each input is then transformed by applying learned weights and biases. This transformation is followed by applying the ReLU activation function, which introduces non-linearity, enabling the model to capture more complex patterns in the data. During training, the network's predictions are evaluated against actual targets using a cross-entropy loss function, augmented with L2 regularization to prevent overfitting. The model employs the gradient descent optimization method for backpropagation, where it adjusts its weights and biases to minimize the loss. Once trained, the network applies these learned parameters to new inputs, generating outputs that represent its predictions.

5.2 Experiment 2

5.2.1 CNN

In this implementation, a Convolutional Neural Network (CNN) is constructed using TensorFlow's Sequential model in which a linear stack of layers is created. The network begins with a 2D convolutional layer (Conv2D) that has 64 filters of size 3x3, and uses 'tanh' activation for non-linear feature extraction. These images are structured with specified dimensions of 100x100 and 3 color channels. Following this, a Max Pooling layer (MaxPooling2D) with a 2x2 window reduces the spatial dimensions of the output, helping in reducing computation and controlling overfitting. A Dropout layer with a rate of 0.01 is added to further mitigate overfitting by randomly dropping out nodes during training.

The network then flattens the output using a Flatten layer, necessary for transforming 2D feature maps into a 1D vector before passing it through dense layers. It includes two dense layers, the first with 32 units and a 'sigmoid' activation function, followed by a final dense layer with 3 units, also with 'sigmoid' activation, corresponding to the number of classes in the classification task. The model is then compiled using the Stochastic Gradient Descent (SGD) optimizer and 'categorical_crossentropy' loss function, suitable for multi-class classification, with accuracy as the performance metric.

The training part involves fitting the model to the training data for 10 epochs, validated against a test dataset. To enhance training efficiency and prevent overfitting, an Early Stopping callback is employed. It monitors the validation accuracy and halts the training if no improvement is observed after 1 epoch, ensuring the model trains only as long as it is making significant improvements on the validation data.

5.2.2 InceptionNet

In its implementation, the model utilises the InceptionV3 architecture as its base. The InceptionV3 model, loaded with weights pre-trained on the ImageNet dataset, is set up without its top layers (using include top=False). This configuration allows the model to be adapted to the specific needs of the task at hand, which involves input images of size 100x100.

The output from the base InceptionV3 model is then passed through a Global Aver-age Pooling 2D layer (GlobalAveragePooling2D). This layer reduces each feature map to a single value, effectively condensing the features extracted by InceptionV3 while maintaining their spatial integrity. Following this, a Dense layer with 128 units and 'relu' activation is added. This layer serves as a fully connected layer that processes the pooled features, allowing for non-linear transformations and learning of more complex patterns.

The final layer in the model is another Dense layer, this time with 3 units and a 'sigmoid' activation function. This layer corresponds to the number of output classes and is responsible for generating the final predictions.

The entire model is then compiled using the 'Adam' optimizer and 'categorical crossentropy' as the loss function.

An important aspect of this implementation is setting all layers in the base InceptionV3 model as trainable (layer. Trainable = True). This approach allows for fine-tuning the entire network, including the pre-trained layers, which can lead to better performance as the model adapts more comprehensively to the specific dataset being used.

An Early Stopping callback is also used during the implementation. The EarlyStopping callback monitors the model's validation accuracy. It checks the performance of the model on a validation dataset after each training epoch. The patience parameter is set to 1 to stop the training if there's no improvement in the validation accuracy for 1 consecutive epoch.

5.2.3 DenseNet121

For the implementation the DenseNet121 is configured without its top layer and is augmented with ImageNet pre-trained weights. The output from DenseNet121 undergoes flattening to transform 2D feature maps into a 1D vector. This is followed by adding a fully connected dense layer with 256 units and 'relu' activation for non-linear processing, and a dropout layer with a 0.02 rate to mitigate overfitting. The final layer is a dense layer with 3 units and a 'tanh' activation.

The model is then compiled using the 'adam' optimizer and 'categorical crossentropy' loss function with accuracy as the evaluation metric. To enhance training efficiency and prevent overfitting, an EarlyStopping callback is implemented, monitoring validation accuracy and halting training if no improvement is observed for one epoch.

5.2.4 VGG19

Initially configured with ImageNet pre-trained weights the top layers of the VGG19 model are excluded for customization. Fine-tuning is achieved by unfreezing the last few layers, allowing them to adjust during training to better suit the dataset. The model then incorporates a Global Average Pooling 2D layer to condense feature maps, followed by a Flatten layer to transform the output into a 1D array. A dense layer with 512 units and 'relu' activation is added for complex pattern recognition, followed by a final output layer with 3 units using 'softmax' activation, suitable for multi-class classification. Compiled with the 'adam' optimizer and 'binary crossentropy' loss function, and monitored for accuracy, the model includes an EarlyStopping callback to enhance training efficiency. This callback halts training if there's no improvement in validation accuracy over one epoch, preventing overfitting and ensuring the model remains efficient and effective for the classification task at hand.

6 Evaluation

This chapter of the report details the results obtained for both the implemented experiments. It critically assesses the findings of the study and provide the complexities associated with the study.

6.1 Experiment / Case Study 1

Table below depicts the result of the models implemented in Experiment 1 on the Breast Cancer Wisconsin Dataset.

Table 7: Result Table

Model	Accuracy (%)	F1-Score (%)	Precision (%)	Recall (%)
Logistic Regression ML	98.88	98.91	100.00	97.85
Neural Network ML	98.32	98.38	98.91	97.85
Logistic Regression JAX	97.77	97.85	97.85	97.85
Neural Network JAX	49.72	51.18	87.97	49.72

The evaluation of the four models presents a varied performance. The Logistic Regression model implemented using the Scikit Learn library stands out with exceptionally high scores across all metrics, including a perfect precision of 100%. This suggests an excellent predictive capability, particularly in correctly identifying positive cases without false positives. However, its slightly lower recall than precision indicates a minor short-fall in capturing all positive instances. The near-perfect F1-score reinforces its overall balanced performance between precision and recall, indicating a robust model for this particular dataset.

In contrast, the Neural Network model implemented using Scikit Learn, while still performing admirably with an accuracy above 98%, shows a slight drop in effectiveness compared to the Logistic Regression. This drop might imply that for the given dataset, the added complexity of a neural network does not translate into significantly better performance. The balance between precision and recall remains strong, as reflected in the F1-score, suggesting that the model is still proficient in handling the classification task.

The Logistic Regression JAX model, though slightly trailing behind its Scikit Learn counterpart, shows competent performance with uniform scores in precision and recall. This balance indicates a consistent performance for both positive and negative classes. The slightly lower scores across the board compared to Logistic Regression might be due to the differences in the implementation or optimization techniques of the JAX framework.

The Neural Network JAX model, however, significantly underperforms in comparison to the other models, with accuracy and F1-score near 50%. The notably low recall suggests a substantial number of positive instances are being incorrectly classified, which is concerning for any classification task. This underperformance could stem from a variety of factors including suboptimal network architecture, insufficient training, or the need for more precise hyperparameter tuning in the JAX environment.

6.2 Experiment / Case Study 2

Model	Accuracy (%)
CNN	56.51
InceptionNet	30.79
Dense Net	56.51
VGG19	76.51

Table 8: Result table

The evaluation of four different models on the Breast Ultrasound Image Dataset paints a different picture. The VGG19 model stands out with the highest accuracy at 76.51%, maybe because of its deep and wide-ranging feature extraction capabilities. However, its computational intensity could be a limiting factor in resource-constrained environments. CNN and Dense Net both exhibit moderate performances with an accuracy of 56.51%, suggesting that while their respective architectures are somewhat effective, there may be room for optimization in terms of network depth, filter sizes, or regularization techniques. InceptionNet, despite its efficient design, trails with the lowest accuracy at 30.79%. This could be attributed to factors such as insufficient fine-tuning, inadequate training epochs, or a mismatch between the model’s architectural strengths and the dataset’s characteristics.

The significant variance in performances indicates that the models' suitability varies considerably with the task specifics.

6.3 Discussion

These findings reveal significant insights into model optimization and data-model alignment in the context of machine learning for medical diagnostics. Specifically, the under-performance of the JAX-based Neural Network in the first experiment poses questions about optimal network architecture and parameter settings within JAX's computational framework. This challenge highlights the need for meticulous hyperparameter tuning and potentially extended training periods, particularly for CNN and Dense Net models, which demonstrated only moderate performance in the second experiment. Experimentation with various layer configurations, activation functions, and advanced techniques like batch normalization could be pivotal in enhancing model efficacy.

The study brings to light the trade-offs between computational efficiency and accuracy. The high accuracy of the computationally demanding VGG19 model raises essential considerations regarding the feasibility and justification of resource-intensive models in clinical settings, where efficiency and quick turnaround are as crucial as accuracy.

7 Conclusion and Future Work

The study looks closely at how machine learning, especially JAX-based models, can be used to find breast cancer. The results of the tests using both structured and image datasets give us useful information about the pros and cons of different models, such as basic machine learning methods and more complex deep learning architectures like CNN, InceptionNet, Dense Net, and VGG19.

One of the most important conclusions is how important model optimization is in medical diagnostic machine-learning applications. The study shows how difficult it is to put complex models like JAX-based Neural Networks into practice. It also stresses how important it is to carefully choose the network architecture, learning rates, and iteration numbers.

It is crucial to ensure that the model architecture is appropriate for the dataset. It has been revealed through a study that the features of a dataset can significantly impact the performance of a model. Therefore, it is crucial to select and tailor models according to the specific requirements of each dataset. This alignment is particularly important to ensure the appropriate use of machine learning tools in medical diagnosis.

The study highlights the challenge of striking a balance between the speed and accuracy of a model and its associated costs. For instance, the VGG19 model's exceptional ac-curacy raises pertinent questions about utilizing resources effectively and ensuring smooth functioning in medical settings where prompt and accurate diagnoses hold utmost significance.

This study demonstrates how machine learning has the potential to revolutionize the way breast cancer is diagnosed. Advanced computer tools like JAX can provide more ac-curate, non-invasive, and faster diagnostic methods, leading to significant improvements in healthcare. However, it also highlights the complexity and challenges in implementing these new technologies in medicine, indicating the need for further research and develop-mint in this field.

7.1 Future Work

In the future, the study of JAX-based machine learning models for detecting breast cancer points towards three important directions that can be further explored. Firstly, there is a significant opportunity to enhance model optimization and diversity by focusing on improving the architectures within the JAX framework and using these models on a wider range of datasets. The goal of this approach is to increase accuracy and flexibility in various clinical situations. Secondly, it is vital to create models that are both technically sound and practical in real life. Using these models in real clinical trials could provide valuable information on how they can be used in reality. Finally, making computers faster, especially for complex models like VGG19, and applying these AI techniques to other areas of healthcare, such as personalized medicine and different types of cancer, could revolutionize how healthcare is diagnosed and treated, putting AI's potential to transform the medical industry at the forefront of innovation.

References

Afaya, A., Ramazanu, S., Bolarinwa, O. A., Yakong, V. N., Afaya, R. A., Aboagye, R. G., Daniels-Donkor, S. S., Yahaya, A.-R., Shin, J. and D, V. M. (2022). Health system barriers influencing timely breast cancer diagnosis and treatment among women in low and middle-income asian countries: evidence from a mixed-methods systematic review, BMC Health Services Research 22(1): 1–17.

Azevedo, A. and Santos, M. (2008). Kdd, semma and crisp-dm: A parallel ove rview, IADS-DM .

Desai, M. and Shah, M. (2021). An anatomization on breast cancer detection and dia-gnosis employing multi-layer perceptron neural network (mlp) and convolutional neural network (cnn), Clinical eHealth 4: 1–11.

Hamed, S., Mesleh, A. and Arabiyyat, A. (2021). Breast cancer detection using machine learning algorithms, International Journal of Computer Science and Mobile Computing 10(11): 4–11.

Kumar, N., Sharma, G. and Bhargava, L. (2020). The machine learning-based optimized prediction method for breast cancer detection, 2020 4th International Conference on Electronics, Communication and Aerospace Technology (ICECA), IEEE, pp. 1594– 1598.

Kumar, U. K., Nikhil, M. S. and Sumangali, K. (2017). Prediction of breast cancer using voting classifier technique, 2017 IEEE International Conference on Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials (ICSTM), IEEE, pp. 108–114.

Lange, R. (2023). Evosax: Jax-based evolution strategies, Proceedings of the Companion Conference on Genetic and Evolutionary Computation pp. 659–662.

Mangukiya, M., Savani, M. and Vaghani, A. (2022). Breast cancer detection with ma-chine learning, International Journal for Research in Applied Science & Engineering Technology (IJRASET) 10(II): 23–30.

Min, S., Heo, J., Kong, Y., Nam, Y., Ley, P., Jung, B., Oh, D. and Shin, W. (2017). Thermal infrared image analysis for breast cancer detection, KSII Transactions on Internet & Information Systems 11(2).

Omondiagbe, D., Veeramani, S. and Sidhu, A. (2019). Machine learning classification techniques for breast cancer diagnosis, IOP Conference Series: Materials Science and Engineering 495: 012033.

Osman, A. (2017). An enhanced breast cancer diagnosis scheme based on a two-step-svm technique, International Journal of Advanced Computer Science and Applications 8(4).

Sardouk, F., Duru, A. and Bayat, O. (2019). Classification of breast cancer using data mining, American Scientific Research Journal for Engineering, Technology, and Sci-ences (ASRJETS) 51(1): 38–46.

Shawarib, M. Z. A., Latif, A. E. A., Al-Zatmah, B. E. E.-D. and Abu-Naser, S. S. (2020). Breast cancer diagnosis and survival prediction using jnn, International Journal of Engineering and Information Systems (IJEAIS) 4(10): 23–30.

Shubham, K. and Kamalraj, R. (2022). Breast cancer detection using machine learning algorithms, International Journal of Advances in Engineering and Management (IJAEM) 4: 987.

Singh, D. and Singh, A. (2020). Role of image thermography in early breast cancer detection - past, present and future, Computer Methods and Programs in Biomedicine 183: 105074.

Wolberg, William, M.-O. S.-N. and Street, W. (1995). Breast Cancer Wisconsin (Diagnostic), UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5DW2B.

Yadav, R., Singh, P. and Kashtriya, P. (2023). Diagnosis of breast cancer using machine learning techniques - a survey, Procedia Computer Science 218: 1434–1443.

Yue, W., Wang, Z., Chen, H., Payne, A. and Liu, X. (2018). Machine learning with applications in breast cancer diagnosis and prognosis, Designs 2(2): 13.