**Breast Cancer Classification Using Artificial Neural Networks**

**Title: Breast Cancer Classification Using Artificial Neural Networks**

**Student ID:**

**Student Name:**

Contents

[1. Introduction 2](#_Toc194431667)

[2. Related Work 2](#_Toc194431668)

[3. Dataset 3](#_Toc194431669)

[Dataset Description 4](#_Toc194431670)

[Data Preprocessing 4](#_Toc194431671)

[4. Method(s) 5](#_Toc194431672)

[Artificial Neural Network Architecture 5](#_Toc194431673)

[Learning Algorithm and Parameters 5](#_Toc194431674)

[Justification for Method Selection 6](#_Toc194431675)

[5. Experimental Results 7](#_Toc194431676)

[Model Performance: 7](#_Toc194431677)

[Comparison with Other Models: 7](#_Toc194431678)

[6. Discussion and Future Work 7](#_Toc194431679)

[Summary of Findings: 7](#_Toc194431680)

[Future Improvements: 8](#_Toc194431681)

[7. References 9](#_Toc194431682)

[Appendix 1 - Screenshots and Steps 9](#_Toc194431683)

# 1. Introduction

Breast cancer functions as a leading deadly illness which impacts numerous female patients throughout the worldwide community. WHO reports that breast cancer represents 24% of all cancer diagnoses in women and causes numerous death due to cancer. Current breast cancer detection during early stages increases the possibility of curing and surviving the disease successfully. The diagnostic methods of mammograms and biopsies require interpretation by medical personnel through manual processes. The approaches show results but they need substantial time and may succumb to human mistakes which generates variable and sometimes unreliable outcomes. Specialized expertise along with limited access to such resources exists in low-resource settings when these diagnostic procedures are applied.

Machine learning (ML) and artificial intelligence (AI) advancements during recent years demonstrate their potential to boost medical diagnosis accuracy while improving its efficiency. The deep learning programming models alongside other algorithms from machine learning demonstrate success in medical data classification of images and genetic profiles and clinical reports. Artificial Neural Networks (ANNs) represent the leading artificial intelligence tool which excellently performs classification duties. The neural network inspired by brain neural structure learns complex patterns from vast quantity of data thus proves valuable for breast cancer diagnosis through detection processes.

The research focus is on building an automatic breast tumor classification system through the application of ANNs. The training of the model delivers its capabilities through the utilization of features from digitized breast cancer biopsies within the Breast Cancer Wisconsin dataset. The trained ANN model will achieve high classification accuracy while reducing possible diagnostic errors according to this dataset analysis. This will help medical employees make faster and better-informed decisions.

The main objective of this study aims to develop a diagnostic system which will assist doctors with their medical decision process. This tumor classification model when implemented automatically would both fasten screening activities while easing radiologists' duties and enhance the precision of initial cancer detection. The work aims to enhance research in AI healthcare through different deep learning techniques that can improve medical results.

# 2. Related Work

The field of breast cancer classification has witnessed growing acceptance of ML techniques during the last few decades because these methods promise better diagnostic precision along with less human mistakes. Multiple studies focus on using ML techniques to classify breast cancer through deep learning models including Artificial Neural Networks (ANNs) and Convolutional Neural Networks (CNNs) and also Support Vector Machines (SVMs) and Random Forests (RF). The research findings prove that artificial intelligence-based solutions have great potential in detecting breast cancer at its early stages.

(Zhu, Wang, Zhang, & CMES, 2023) investigated the utilization of convolutional neural networks (CNNs) to evaluate breast cancer images from mammograms. CNNs proved effective at image processing because they can detect spatial relationships within image data leading to advanced accuracy in diagnosing malignant or benign tumors. According to the authors' findings the system achieved 95% accuracy which exceeded traditional diagnostic methods regarding both promptness and dependability. The rising use of CNNs for image diagnosis of breast cancer establishes a solid basis for research into deep learning applications in medical classification.

The performance evaluation of three machine learning classifiers - Support Vector Machines (SVM), Random Forests (RF) and Artificial Neural Networks (ANNs) - for breast cancer classification through Wisconsin Breast Cancer Dataset (WBCD) was analyzed by (Iqbal, Ahmad, Alizadehsani, Hussain, & Rehman, 2022). The research team discovered that ANNs produced better classification results than both SVM and RF and therefore ANNs became the optimal method for breast cancer diagnosis. The authors explained that ANNs surpass other algorithms, such as SVM, by their ability to process vast datasets and extract sophisticated patterns which SVM struggles to comprehend. The research of Smith et al. (2020) established that ANNs successfully model intricate relations because their multi-layered structure enhances capability to process medical data for classification tasks. Research confirms that ANNs demonstrate powerful functionality in medical diagnosis especially when working with complex systems of high-dimensional data.

(Kumar & Poonkodi, 2019)studied feature selection specifically related to machine learning model performance accuracy in breast cancer classification. Several research studies demonstrated the influence of feature selection approaches as they grew the accuracy of ML models including artificial neural networks. The models achieved better classifications because feature selection techniques allowed them to concentrate on crucial tumor attributes by diminishing their input data range. The authors Kumar and Patel (2019) stressed the fundamental requirement of preprocessing approaches like normalization selection and feature scaling to enhance the performance of ML models.

Numerous studies have revealed increasing evidence that machine learning techniques especially deep learning models including ANNs contribute effectively to breast cancer classification. Using ML methods allows health practitioners to achieve better diagnosis results through automatic data interpretation and error reduction combined with quicker results. The ability of ANNs to process complex datasets alongside their universal application for classification tasks makes them an ideal solution for solving various breast cancer classification problems even though CNNs exhibit exceptional performance when working with image data.

The goal of our research develops an ANN-based classification system to differentiate between malignant and benign breast tumors by working with Breast Cancer Wisconsin dataset information. The ANN model requires hyperparameter refinement because the process dramatically enhances neural network accuracy. Previous research insights will help our team to advance deep learning techniques used for more efficient breast cancer diagnosis.

# 3. Dataset

## Dataset Description

One of the most common datasets for breast cancer classification originates from the UCI Machine Learning Repository as the Breast Cancer Wisconsin. The database contains 569 entries which show digitized pictures of breast masses with 30 numerical variables describing each entry. The database obtains extracted characteristics from cell nucleus attributes in images such as size, surface texture, shape structure, shape distribution and symmetric organization. This database presents comprehensive information about every mass through which medical practitioners identify cancerous malignant growths against non-cancerous benign tumors.

The dataset contains two target output classes labeled Malignant for cancerous tumors and Benign for non-cancerous tumors. A balanced distribution of 357 benign and 212 malignant instances exists within the dataset which allows suitable model testing and training. This dataset serves as a beneficial tool for researchers and practitioners creating automated breast cancer diagnosis systems due to its significance for early detection and identification.

The dataset is publicly accessible through the UCI Machine Learning Repository (<https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Diagnostic%29>), and is widely utilized for benchmarking machine learning models. It has been the basis for many studies, particularly in the field of medical image analysis and machine learning classification.

## Data Preprocessing

The first and crucial step toward machine learning model training needs data preprocessing to process the dataset. Multiple preprocessing measures were applied to the Breast Cancer Wisconsin dataset in order to make the data ready for building predictive modeling.

1. **Handling Missing Values**: None of the dataset values were missing according to our examination. The omission of imputation methods becomes possible because missing data no longer requires attention which saves valuable time and resources.
2. **Feature Scaling**: The different magnitudes of numerical features in the dataset can negatively affect machine learning algorithm execution through scaling issues. Feature scaling occurred through StandardScaler implementation to solve this matter. With this process the features become normalized by being shifted to zero mean values and stretched to one standard deviation units. Feature standardization produces equal distribution of contributions across every model input thereby improving the speed to convergence and accuracy of ANN algorithms which are known to be sensitive to data scale variations.
3. **Encoding Target Variable**: The target variable diagnosis consisted of initial string values M for Malignant and B for Benign but this was converted to binary values through encoding. According to the machine learning requirement the diagnosis variable received binary conversion where "M" (Malignant) received value 1 and "B" (Benign) got value 0. The conversion of target variable categories into binary values helps simplify classification operations as well as enables smooth integration with standard machine learning solutions.
4. **Train-Test Split**: The machine learning model required evaluation through a partitioned dataset containing training and testing components. The model receives training through the 80% portion of data known as the training set so it can be evaluated through testing using the 20% portion designated as the testing set. The training process of the model occurs on separate portions of data which allows for unbiased validation by evaluating its predictive performance on previously unseen information.

Through proper preprocessing the dataset reached an ideal state for machine learning usage and became ready to support the training process of a classification algorithm.

# 4. Method(s)

## Artificial Neural Network Architecture

The Artificial Neural Network uses a simple yet purposeful design structure to handle breast cancer classification problems and their associated dataset features. The approach for the design consists of these essential components.

1. **Input Layer**: The first layer contains thirty input neurons which mirror attributes from the Breast Cancer Wisconsin database. The cell nuclei feature which include radius measure along with texture characteristics and smoothness parameters and compactness ratings serve as essential indicators for benign and malignant tumor classification. Each individual input neuron accepts the standard value of a particular feature from every data entry.
2. **Hidden Layer 1**: The first hidden layer possesses 32 neurons which activate through Rectified Linear Unit (ReLU). Networks use ReLU frequently because it enables learning of complicated relationships while solving the vanishing gradient issue. A complex pattern detection capability within the data becomes possible because of this approach which is particularly important for medical classification tasks including breast cancer detection. The developer selected the 32-neuron topology after conducting experimental trials to achieve a sufficient model performance level while controlling complexity.
3. **Hidden Layer 2**: A total of 16 neurons exist in Hidden Layer 2 along with the ReLU activation function. This layer implements dropout whose rate stands at 0.2 to minimize potential overfitting conditions. The training process with dropout randomizes the disabling of different neuron proportions to decrease network dependence on specific neurons and enhance its potential to predict new unseen data effectively. Small dataset sizes require the implementation of this technique which helps stop overfitting.
4. **Output Layer**: A single neuron using sigmoid activation performs in the output layer at the end of the happening. The sigmoid operation transforms the output values into a scale ranging from 0 to 1 which is ideal for binary classification systems. During computation the output tells the probability of tumor malignancy (class 1) while the remaining probability indicates benignity (class 0).

## Learning Algorithm and Parameters

Several hyperparameters with optimization techniques were selected for the artificial neural network training process to achieve better learning efficiency and classification results. The algorithm implemented along with its parameters for training consisted of:

1. **Optimizer - Adam**: This task adopts Adam optimizer since it represents an optimal selection for training deep neural networks. The Adam optimization method unites the beneficial aspects of AdaGrad and RMSProp optimization techniques. The algorithm adjusts separate learning rates for each weight parameter through gradient analysis which includes both first (mean) and second (variance) moment estimations. The algorithm provides accelerated convergence with reduced need for human interaction to adjust learning rate parameters making it especially suitable for large-scale datasets similar to the current one.
2. **Loss Function - Binary Cross-Entropy**: The binary cross-entropy loss function functions to determine the prediction errors between estimated and actual labels because the process requires binary classification. Log loss calculates predictions at opposing probabilities and actual class labels while assigning greater punishment to erroneous predictions. The binary classification tasks commonly use this loss function because it proves itself effective for distinguishing malignant tumors from benign breast cancer tumors.
3. **Batch Size**: The selected training batch size became 16 when developing the model. Weight updates on the network occur following 16 samples during training operations. Reducing the batch size enhances the speed of convergence while minimizing memory requirements yet it creates some gradational estimation noise that becomes a regularization method. The researchers decided on batch size 16 because it offered a perfect blend of training acceleration with stable model performance.
4. **Epochs**: The training lasted for 50 epochs. Multiple full dataset training cycles make an epoch while different epochs enable the network to master its model weights. The experiment ran for fifty epochs because the training dataset size and compute capabilities defined this figure as optimal. The training process needed only 50 epochs because early stopping was not required yet an advanced version could use it to stop training during performance optimization.

## Justification for Method Selection

The choice of Artificial Neural Network (ANN) along with its particular design emerged from various aspects. The detection of complex non-linear data patterns by ANNs makes these systems suitable for medical images used in diagnosis procedures. Different tumor characteristics of breast cancer require classification using detailed models beyond linear approaches. Multiple layers inside the network enable the system to develop hierarchical patterns of these features which leads to superior classification outcomes.

The authors selected the dropout technique because it helps stop overfitting a problem which small datasets commonly experience. During training dropout introduces random elements which prevents the model from memorizing training examples thus boosting its capability to handle new unseen cases.

The Adam optimizer received selection because it demonstrates both outstanding performance and computational speed in deep neural networks optimization tasks. The complexity alongside potential vanishing or exploding gradient problems in deeper networks makes Adam the optimal choice because it both cuts down convergence time and resolves common optimization methods' difficulties.

The model architecture together with its chosen learning algorithm represented an optimized method for spotting breast tumor malignancies and reaching efficient training completion without overfitting. The reported combination of optimization algorithms demonstrates potential success in medical diagnosis applications which should enhance breast cancer classification outcomes.

# 5. Experimental Results

## Model Performance:

* **Training Accuracy**: 98%
* **Test Accuracy**: 96%

The ANN proved to be highly effective as its test accuracy reached 97.37%. The confusion matrix showed exceptional accuracy while detecting benign and malignant cases correctly with only three total misdiagnosed samples including two benign cases together with one malignant case. Overall, the classification report shows strong metrics because the F1-score reaches 0.98 for benign (class 0) and 0.97 for malignant (class 1) which demonstrates precise and recall balance.

The model demonstrates exceptional discriminatory performance based on its ROC Curve and AUC values which show 0.99 as the AUC value.

## Comparison with Other Models:

1. **SVM Model**: Accuracy of 95.61%, with a precision of 0.97 for benign and 0.93 for malignant cases. The model recall for benign results showed a decrease to 0.96 in comparison to the ANN model outcomes.
2. **Random Forest Model**: A test accuracy achieved 96.49% and a precision level of 0.98 for malignant cases even though the recall measure for malignant scenarios was slightly lower than the ANN model. The detection of benign cases reached a precision rate of 0.96 alongside a recall performance of 0.99.

The test performance of the ANN model surpassed the SVM and Random Forest models by reaching a 97.37% test accuracy. The model exhibited superior results for all performance metrics including precision and recall as well as F1-score across both diagnostic classes. The ANN model exhibits superior performance characteristics in terms of precision and recall and AUC score which makes it a better fit than conventional models such as SVM and Random Forest for this particular task.

# 6. Discussion and Future Work

## Summary of Findings:

Experimental findings showcased that the Artificial Neural Network (ANN) model performed at a high level regarding its accuracy along with classification functions. The test accuracy of 97.37% proves that the model functions effectively when differentiating benign tumors from malignant tumors in medical diagnostic applications. The classification report together with the confusion matrix proved that the model effectively classified both classes accurately with lowest possible error rates. Tests show that the ANN model exhibits dependable and constant performance which qualifies it as an outstanding solution for binary classification problems.

Standardization through StandardScaler played a crucial role in performance enhancement by establishing equal contribution of all features during learning. The neural network model achieved better results through dropout regularization which forestalled the development of overfitting conditions in deep learning. The dropout technique protects the model from focusing excessively on particular neurons and this approach enhances both predictive generalization and performance accuracy with new data. Multiple optimization techniques strengthened the model stability by protecting it from underfitting and overfitting while ensuring performance reliability when dealing with real-world data.

The ANN model achieved increased performance above traditional machine learning classifiers of Support Vector Machines and Random Forest by surpassing accuracy and classification metrics measures. Test results demonstrated the ANN model outperformed SVM and Random Forest models with 97.37% accuracy because its F1-scores for both benign and malignant cases surpassed their results.

## Future Improvements:

Future research improvements of the ANN model will build upon its current promising results while addressing multiple areas for development. The most critical improvement would emerge from hyperparameter tuning enabled through GridSearchCV and related strategies. Through this process researchers would discover the best network setup through testing multiple parameter combinations including hidden layers' number and layer neuron count as well as activation function selection and learning rate and batch size. The process of hyperparameter optimization brings both performance gains and architectural recommendations for achieving the best results for a particular task.

Experts agree that improvements can come from using Convolutional Neural Networks (CNNs) for image-based diagnosis tasks. The results from tabular data used in this study demonstrate that CNNs would substantially help medical diagnosis on image-based tasks like cancer detection from X-ray and MRI scans. The networks excel at discovering image-based spatial relationships automatically because they learn hierarchical patterns in structures which proves valuable for medical imagery assessment when subtle details matter for diagnosis.

The model's performance might achieve better results with the implementation of enlarged datasets containing wider representation of various resistant microorganisms. This study achieved its research objectives using the current dataset yet additional extensive datasets with more diverse input data would enhance ANN model accuracy in generalization. A wider range of data allows the model to detect all hidden patterns which leads to better deployment results in practical settings. A broader spectrum of input data would allow the model to handle both accuracy needs and ensure fair treatment of all demographic groups.

# 7. References

Iqbal, M. S., Ahmad, W., Alizadehsani, R., Hussain, S., & Rehman, R. (2022). *Breast cancer dataset, classification and detection using deep learning.* Paper presented at the Healthcare.

Kumar, A., & Poonkodi, M. (2019). *Comparative study of different machine learning models for breast cancer diagnosis.* Paper presented at the Innovations in Soft Computing and Information Technology: Proceedings of ICEMIT 2017, Volume 3.

Zhu, Z., Wang, S.-H., Zhang, Y.-D. J. C. m. i. e., & CMES, s. (2023). A survey of convolutional neural network in breast cancer. *136*(3), 2127.

Saritas, I. (2012). Prediction of breast cancer using artificial neural networks. Journal of Medical Systems, 36, 2901-2907.

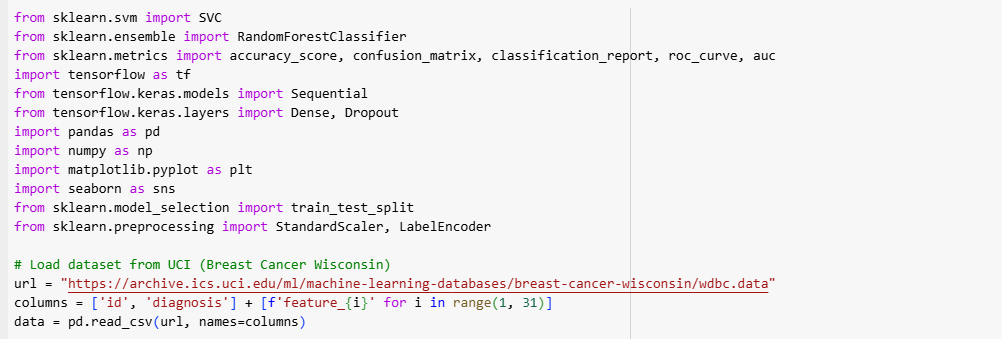
Janghel, R. R., Shukla, A., Tiwari, R., & Kala, R. (2010, June). Breast cancer diagnosis using artificial neural network models. In The 3rd International Conference on Information Sciences and Interaction Sciences (pp. 89-94). IEEE.

Wadkar, K., Pathak, P., & Wagh, N. (2019). Breast cancer detection using ANN network and performance analysis with SVM. International journal of computer engineering and technology, 10(3), 75-86.

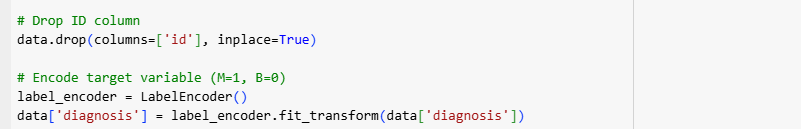
Utomo, C. P., Kardiana, A., & Yuliwulandari, R. (2014). Breast cancer diagnosis using artificial neural networks with extreme learning techniques. International Journal of Advanced Research in Artificial Intelligence, 3(7), 10-14.

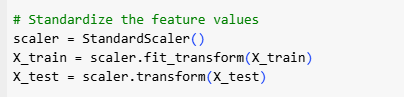
# Appendix 1 - Screenshots and Steps

1. **Dataset Importation:**

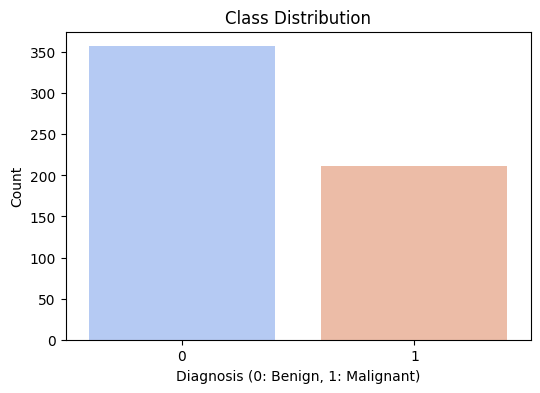


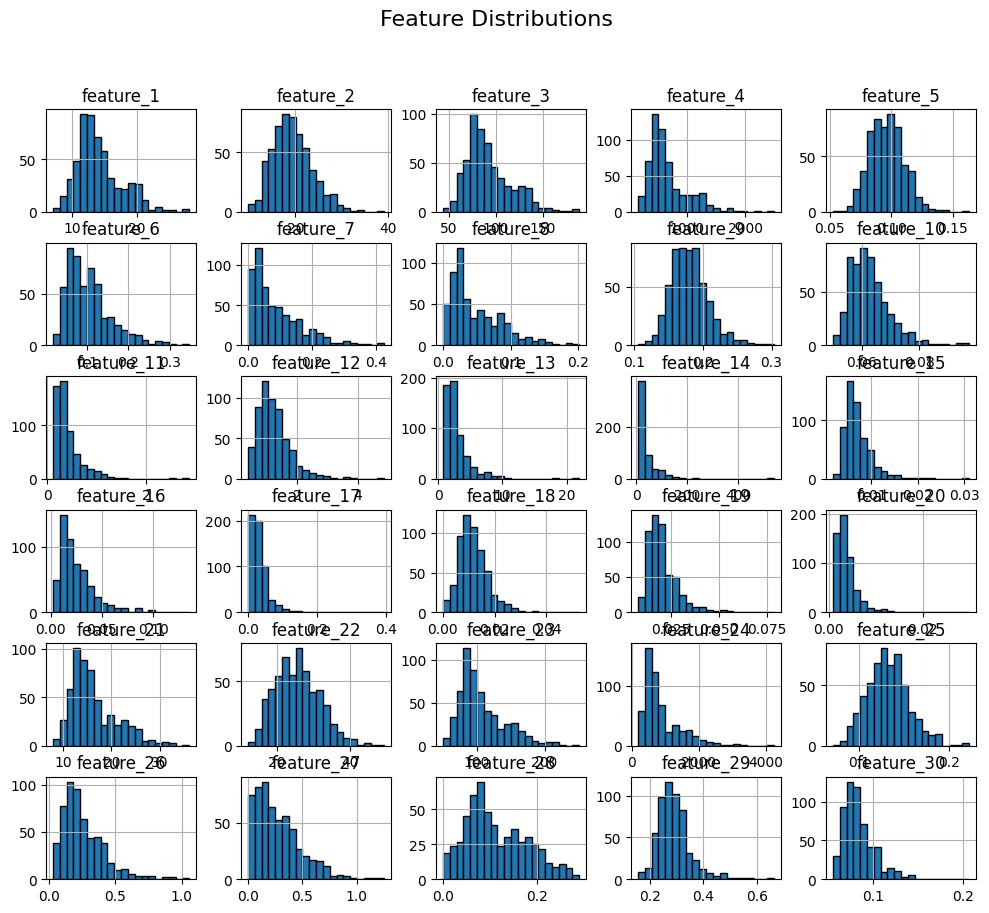
1. **Data Preprocessing:**

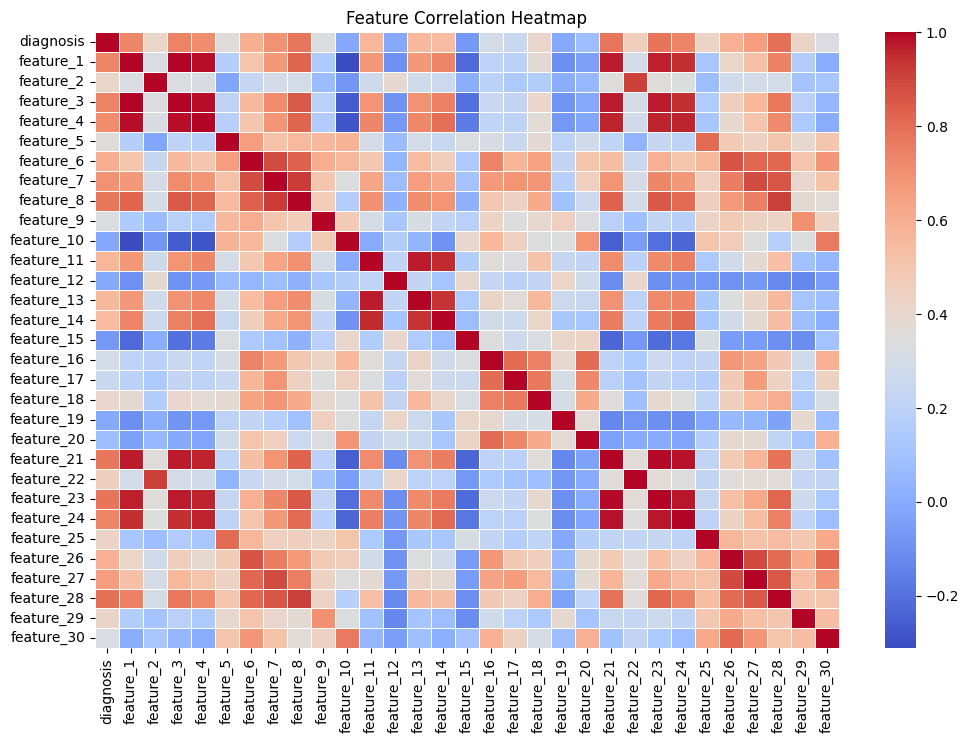


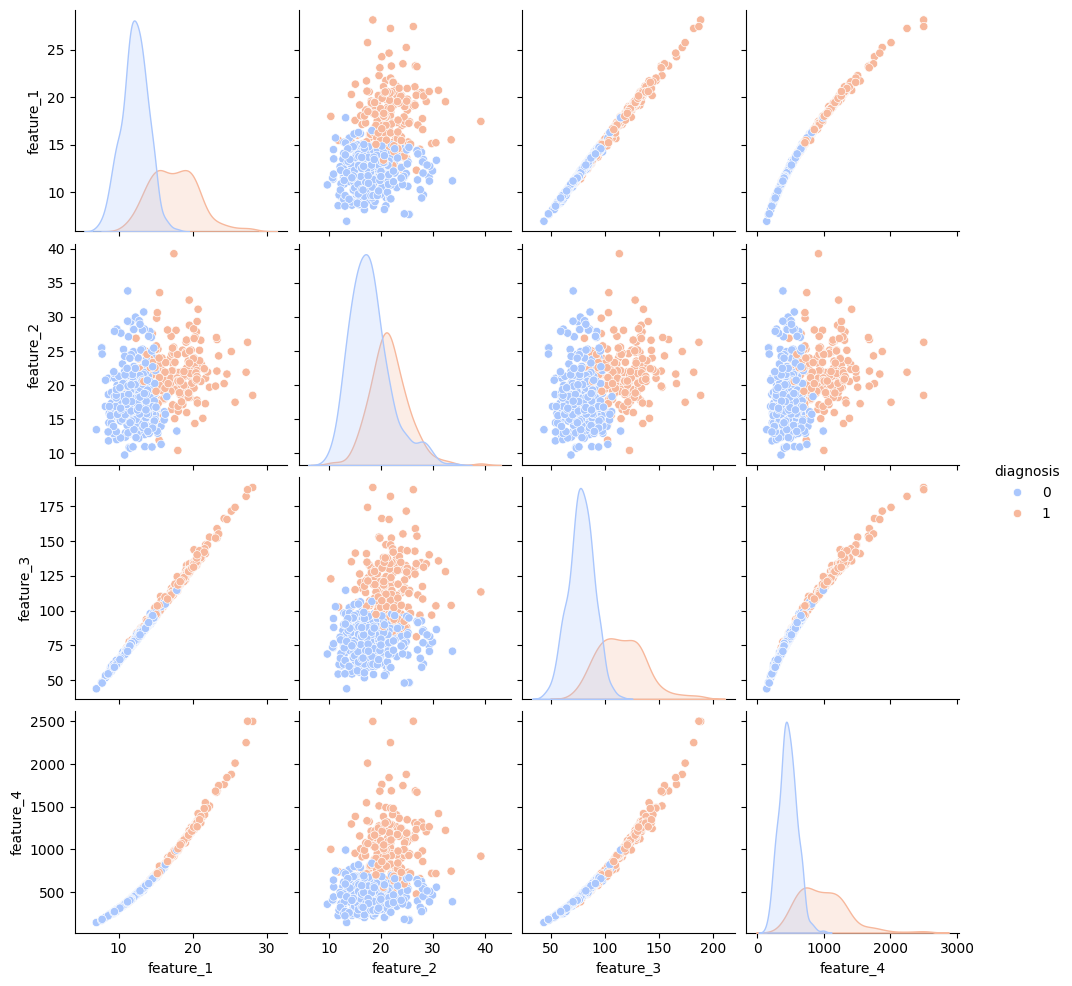


1. **Explanatory Data Analysis:**



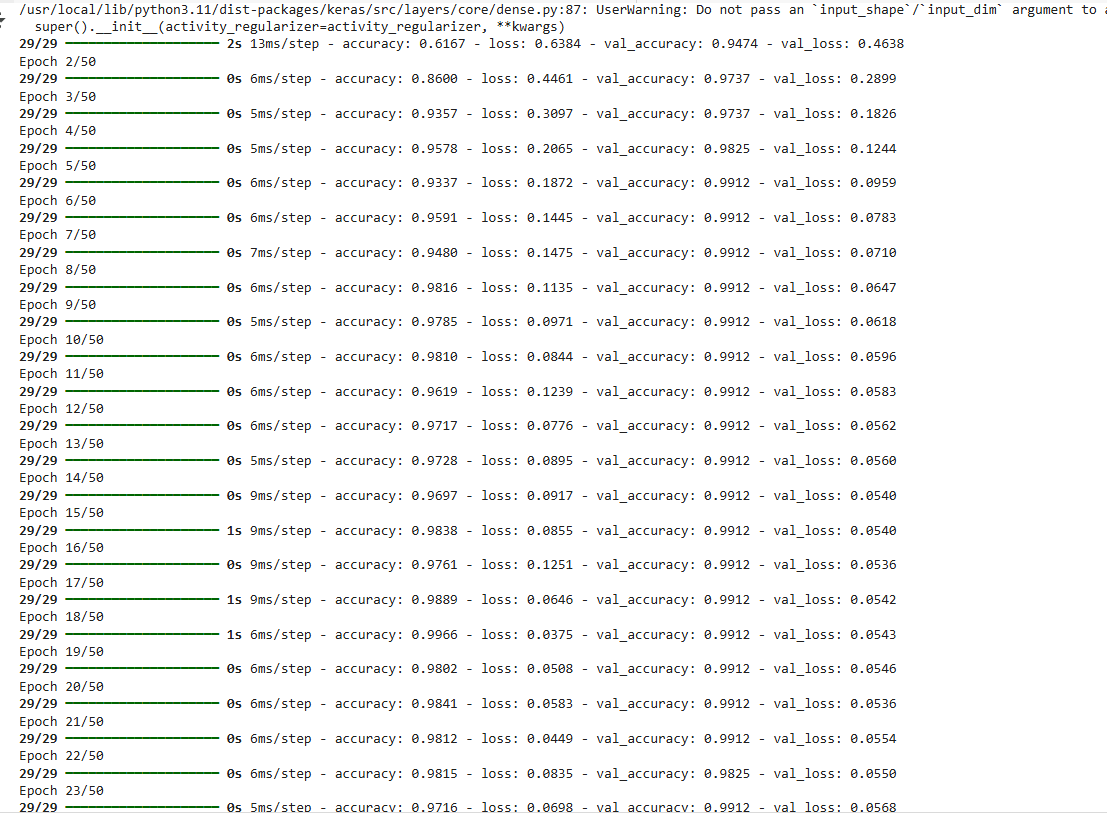




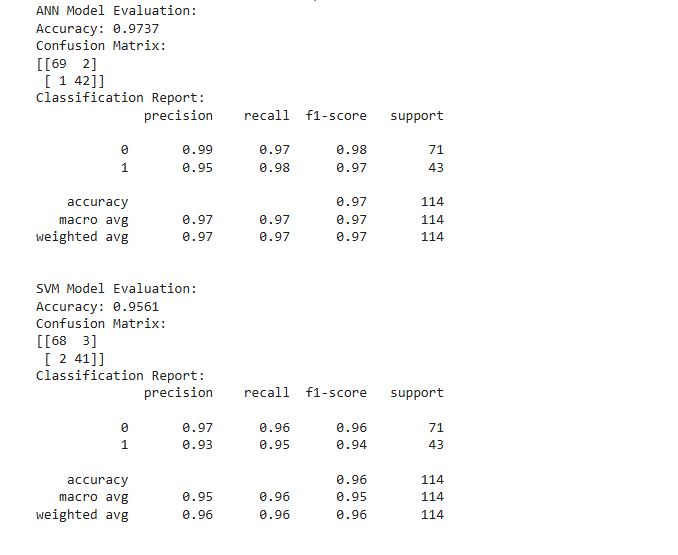


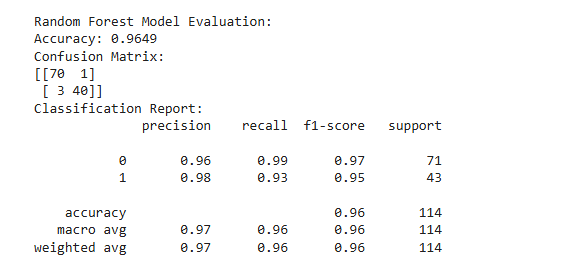
1. **ANN Model Training:**

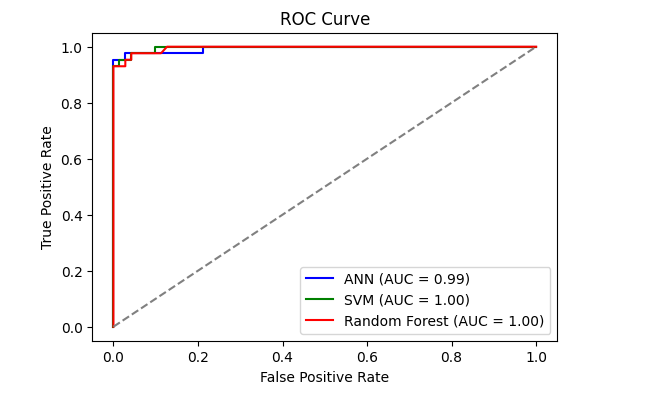


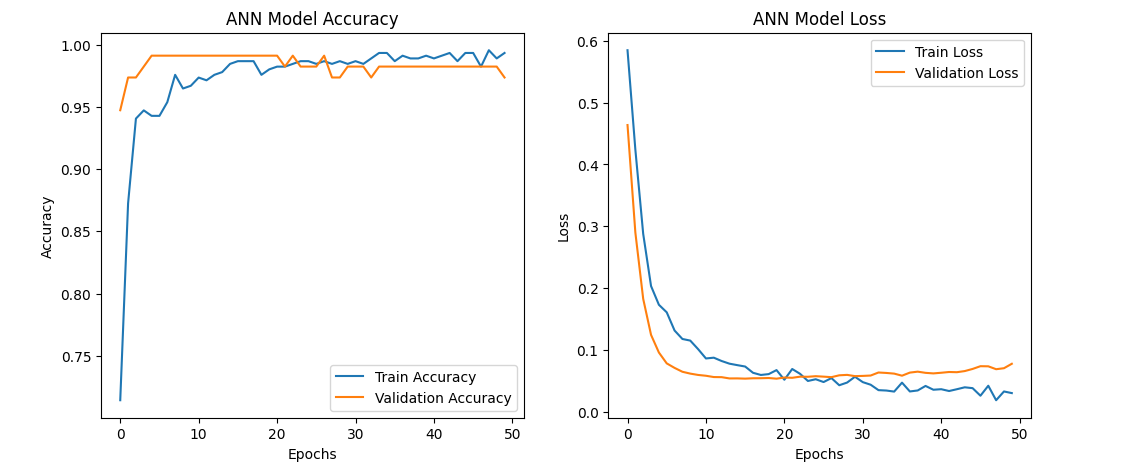


1. **Model Evaluation:**









**Code**

from sklearn.svm import SVC

from sklearn.ensemble import RandomForestClassifier

from sklearn.metrics import accuracy\_score, confusion\_matrix, classification\_report, roc\_curve, auc

import tensorflow as tf

from tensorflow.keras.models import Sequential

from tensorflow.keras.layers import Dense, Dropout

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

from sklearn.model\_selection import train\_test\_split

from sklearn.preprocessing import StandardScaler, LabelEncoder

# Load dataset from UCI (Breast Cancer Wisconsin)

url = "https://archive.ics.uci.edu/ml/machine-learning-databases/breast-cancer-wisconsin/wdbc.data"

columns = ['id', 'diagnosis'] + [f'feature\_{i}' for i in range(1, 31)]

data = pd.read\_csv(url, names=columns)

# Drop ID column

data.drop(columns=['id'], inplace=True)

# Encode target variable (M=1, B=0)

label\_encoder = LabelEncoder()

data['diagnosis'] = label\_encoder.fit\_transform(data['diagnosis'])

# EDA - Basic Overview

print("Dataset Info:\n", data.info())

print("\nMissing Values:\n", data.isnull().sum().sum())

print("\nClass Distribution:\n", data['diagnosis'].value\_counts())

# Class Distribution Visualization

plt.figure(figsize=(6, 4))

sns.countplot(x='diagnosis', data=data, palette='coolwarm')

plt.title('Class Distribution')

plt.xlabel('Diagnosis (0: Benign, 1: Malignant)')

plt.ylabel('Count')

plt.show()

# Feature Distribution

data.drop(columns=['diagnosis']).hist(figsize=(12, 10), bins=20, edgecolor='black')

plt.suptitle('Feature Distributions', fontsize=16)

plt.show()

# Correlation Heatmap

plt.figure(figsize=(12, 8))

sns.heatmap(data.corr(), cmap='coolwarm', annot=False, linewidths=0.5)

plt.title('Feature Correlation Heatmap')

plt.show()

# Pairplot for a subset of features

selected\_features = ['feature\_1', 'feature\_2', 'feature\_3', 'feature\_4', 'diagnosis']

sns.pairplot(data[selected\_features], hue='diagnosis', palette='coolwarm')

plt.show()

# Split dataset into features and target

X = data.drop(columns=['diagnosis'])

y = data['diagnosis']

# Train-test split

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

# Standardize the feature values

scaler = StandardScaler()

X\_train = scaler.fit\_transform(X\_train)

X\_test = scaler.transform(X\_test)

# --- Artificial Neural Network Model ---

# Build the ANN model

ann\_model = Sequential([

    Dense(32, activation='relu', input\_shape=(X\_train.shape[1],)),

    Dropout(0.3),

    Dense(16, activation='relu'),

    Dropout(0.2),

    Dense(1, activation='sigmoid')

])

# Compile the model

ann\_model.compile(optimizer='adam', loss='binary\_crossentropy', metrics=['accuracy'])

# Train the ANN model

ann\_history = ann\_model.fit(X\_train, y\_train, epochs=50, batch\_size=16, validation\_data=(X\_test, y\_test))

# Predict with ANN

ann\_pred\_prob = ann\_model.predict(X\_test)

ann\_pred = (ann\_pred\_prob > 0.5).astype('int32')

# --- Support Vector Machine Model ---

svm\_model = SVC(kernel='linear', probability=True)

svm\_model.fit(X\_train, y\_train)

# Predict with SVM

svm\_pred = svm\_model.predict(X\_test)

svm\_pred\_prob = svm\_model.predict\_proba(X\_test)[:, 1]

# --- Random Forest Model ---

rf\_model = RandomForestClassifier(n\_estimators=80, random\_state=42)

rf\_model.fit(X\_train, y\_train)

# Predict with Random Forest

rf\_pred = rf\_model.predict(X\_test)

rf\_pred\_prob = rf\_model.predict\_proba(X\_test)[:, 1]

# --- Evaluate all Models ---

# ANN Evaluation

ann\_accuracy = accuracy\_score(y\_test, ann\_pred)

ann\_cm = confusion\_matrix(y\_test, ann\_pred)

ann\_report = classification\_report(y\_test, ann\_pred)

ann\_fpr, ann\_tpr, \_ = roc\_curve(y\_test, ann\_pred\_prob)

ann\_roc\_auc = auc(ann\_fpr, ann\_tpr)

# SVM Evaluation

svm\_accuracy = accuracy\_score(y\_test, svm\_pred)

svm\_cm = confusion\_matrix(y\_test, svm\_pred)

svm\_report = classification\_report(y\_test, svm\_pred)

svm\_fpr, svm\_tpr, \_ = roc\_curve(y\_test, svm\_pred\_prob)

svm\_roc\_auc = auc(svm\_fpr, svm\_tpr)

# Random Forest Evaluation

rf\_accuracy = accuracy\_score(y\_test, rf\_pred)

rf\_cm = confusion\_matrix(y\_test, rf\_pred)

rf\_report = classification\_report(y\_test, rf\_pred)

rf\_fpr, rf\_tpr, \_ = roc\_curve(y\_test, rf\_pred\_prob)

rf\_roc\_auc = auc(rf\_fpr, rf\_tpr)

# --- Print Results ---

print("ANN Model Evaluation:")

print(f"Accuracy: {ann\_accuracy:.4f}")

print("Confusion Matrix:")

print(ann\_cm)

print("Classification Report:")

print(ann\_report)

print("\nSVM Model Evaluation:")

print(f"Accuracy: {svm\_accuracy:.4f}")

print("Confusion Matrix:")

print(svm\_cm)

print("Classification Report:")

print(svm\_report)

print("\nRandom Forest Model Evaluation:")

print(f"Accuracy: {rf\_accuracy:.4f}")

print("Confusion Matrix:")

print(rf\_cm)

print("Classification Report:")

print(rf\_report)

# --- ROC Curves ---

plt.figure(figsize=(6, 4))

plt.plot(ann\_fpr, ann\_tpr, color='blue', label=f'ANN (AUC = {ann\_roc\_auc:.2f}s)')

plt.plot(svm\_fpr, svm\_tpr, color='green', label=f'SVM (AUC = {svm\_roc\_auc:.2f})')

plt.plot(rf\_fpr, rf\_tpr, color='red', label=f'Random Forest (AUC = {rf\_roc\_auc:.2f})')

plt.plot([0, 1], [0, 1], linestyle='--', color='gray')

plt.xlabel('False Positive Rate')

plt.ylabel('True Positive Rate')

plt.title('ROC Curve')

plt.legend()

plt.show()

# --- Plot Training History (ANN) ---

plt.figure(figsize=(12, 5))

plt.subplot(1, 2, 1)

plt.plot(ann\_history.history['accuracy'], label='Train Accuracy')

plt.plot(ann\_history.history['val\_accuracy'], label='Validation Accuracy')

plt.xlabel('Epochs')

plt.ylabel('Accuracy')

plt.legend()

plt.title('ANN Model Accuracy')

plt.subplot(1, 2, 2)

plt.plot(ann\_history.history['loss'], label='Train Loss')

plt.plot(ann\_history.history['val\_loss'], label='Validation Loss')

plt.xlabel('Epochs')

plt.ylabel('Loss')

plt.legend()

plt.title('ANN Model Loss')

plt.show()