Project Report

CancerVision: Advanced Breast Cancer Prediction with Deep Learning

Team ID: NM2023TMID17489

Project Report

1. INTRODUCTION

1.1 Project Overview

Breast cancer is one of the main causes of cancer death worldwide. Computer-aided diagnosis systems showed potential for improving the diagnostic accuracy. But early detection and prevention can significantly reduce the chances of death. It is important to detect breast cancer as early as possible. The goal is to classify images into two classifications of malignant and benign. As early diagnostics significantly increases the chances of correct treatment and survival. In this application we are helping the doctors and patients to classify the Type of Tumor for the specific image given with the help of Neural Networks.

1.2 Purpose

The purpose of CancerVision is to advance breast cancer prediction, specifically targeting advanced stages of the disease, through the application of deep learning techniques. The project aims to enable early detection, improve prediction accuracy, support timely intervention, enhance clinical decision-making, ensure accessibility and usability, and contribute to breast cancer research.

Ultimately, CancerVision seeks to improve patient outcomes by providing healthcare professionals with a powerful tool for early breast cancer detection and intervention.

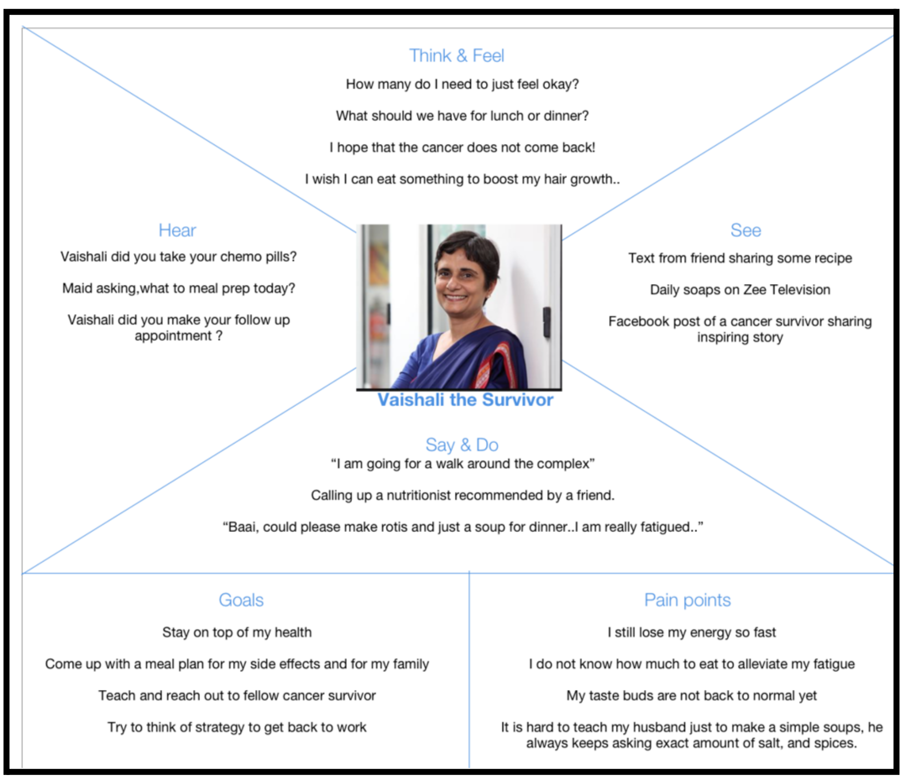
2. IDEATION & PROPOSED SOLUTION

2.1 Problem Statement Definition

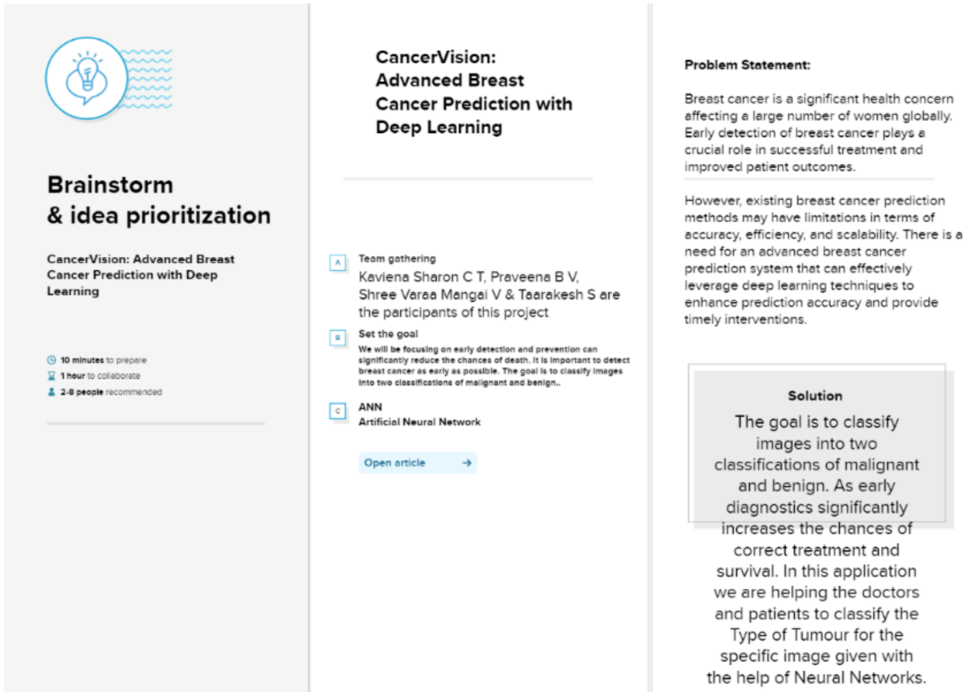
Breast cancer is a significant health concern affecting a large number of women globally. Early detection of breast cancer plays a crucial role in successful treatment and improved patient outcomes.

However, existing breast cancer prediction methods may have limitations in terms of accuracy, efficiency, and scalability. There is a need for an advanced breast cancer prediction system that can effectively leverage deep learning techniques to enhance prediction accuracy and provide timely interventions.

2.2 Empathy Map Canvas



2.3 Ideation & Brainstorming



2.4 Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Current breast cancer prediction methods have limitations in accurately identifying malignancies, leading to false positives or false negatives. The proposed solution aims to improve the accuracy of breast cancer prediction, reducing misdiagnosis rates and ensuring timely intervention.
2.	Idea / Solution description	The goal is to classify images into two classifications of malignant and benign. As early diagnostics significantly increases the chances of correct treatment and survival. In this application we are helping the doctors and patients to classify the Type of Tumour for the specific image given with the help of Neural Networks.

3.	Novelty / Uniqueness	CancerVision stands out for its integration of deep learning, advanced image analysis, scalability, and focus on interpretability. These unique characteristics make it a promising and innovative solution for advanced breast cancer prediction, contributing to improved diagnosis and patient outcomes.
4.	Social Impact / Customer Satisfaction	Overall, CancerVision's social impact lies in improving patient outcomes, enhancing accessibility to healthcare, reducing costs, and empowering healthcare professionals. Its emphasis on accuracy, user-friendliness, time efficiency, and interpretability contributes to customer satisfaction and ensures that the system meets the needs and expectations of healthcare professionals and patients.
5.	Business Model (Revenue Model)	The revenue model for CancerVision will depend on factors such as the competitive landscape, market demand, regulatory considerations, and the value proposition of the predictive system. It's advisable to conduct market research, engage with potential customers, and adapt the revenue model accordingly to ensure sustainability and growth. The total cost estimate for this model would round up to 1.5 Lakh.
6.	Scalability of the Solution	CancerVision can handle increasing demands, accommodate growing datasets, and scale effectively to meet the needs of a broader user base. Scalability is crucial for the widespread adoption and impact of CancerVision in the field of breast cancer prediction and diagnosis.

3.REQUIREMENT ANALYSIS

3.1 Functional requirements

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Data Acquisition and Preprocessing:	Collect relevant medical data, such as patient demographics, medical history, imaging data, genetic information, and biopsy results.
FR-2	Feature Extraction and Selection:	Identify appropriate features from the collected data that are relevant to cancer prediction, such as tumor size, lymph node involvement, hormone receptor status, and genetic mutations.
FR-3	Model Development:	Train the prediction model using labeled data, where the target variable is the presence or absence of cancer.
FR-4	Model Validation and Evaluation:	Conduct rigorous testing and validation of the prediction model using independent datasets or cross-validation techniques.

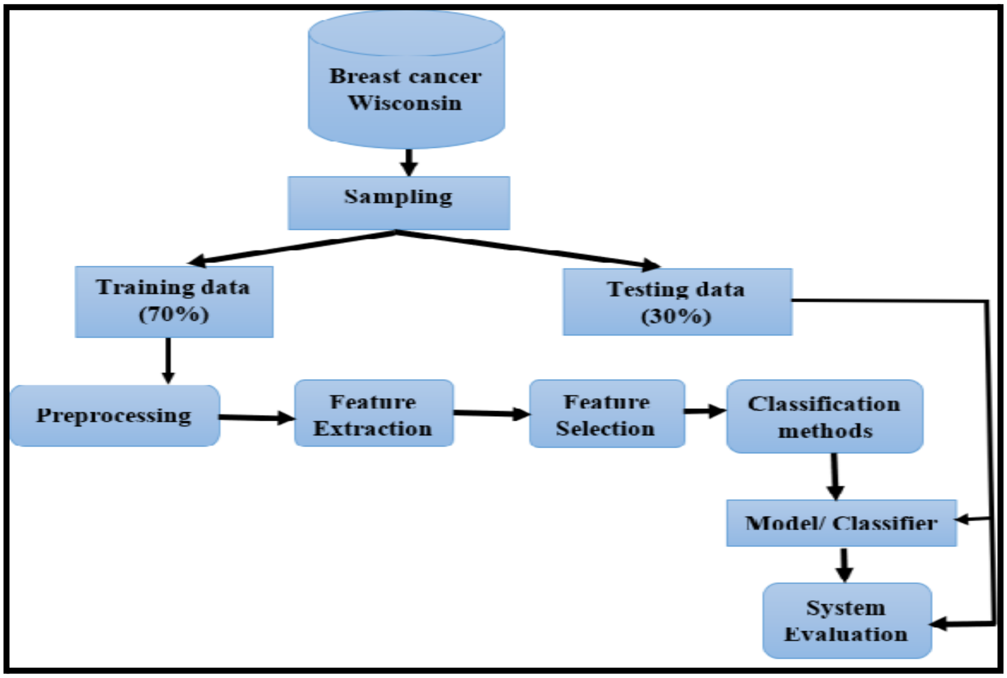
3.2 Non Functional Requirements

Following are the non-functional requirements of the proposed solution

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	The system should provide a user-friendly interface that is intuitive, easy to navigate, and requires minimal training to operate.
NFR-2	Security	The model should employ appropriate security measures to protect against unauthorized access, data breaches, or tampering with the system or data.
NFR-3	Reliability	The model should consistently produce reliable and consistent predictions, exhibiting minimal variability across multiple runs or inputs.
NFR-4	Performance	The model should provide predictions within an acceptable time frame, considering the volume and complexity of the input data.
NFR-5	Maintainability	The system should be designed in a modular and well-structured manner, facilitating easy maintenance, updates, and bug fixes.

4. PROJECT DESIGN

4.1 Data Flow Diagrams of CancerVision: Advanced Breast Cancer Prediction With Deep Learning:



4.2 Solution

(Breast cancer is one of the main causes of cancer death worldwide. Computer-aided diagnosis systems showed potential for improving the diagnostic accuracy. But early detection and prevention can significantly reduce the chances of death. It is important to detect breast cancer as early as possible. The goal is to classify images into two classifications of malignant and benign. As early diagnostics significantly increases the chances of correct treatment and survival. In this application we are helping the doctors and patients to classify the Type of Tumor for the specific image given with the help of Neural Networks.)

- The user interacts with the UI (User Interface) to choose the image.
- The chosen image analyzed by the model which is integrated with flask application.
- CNN Models analyze the image, then prediction is showcased on the Flask UI.

Technical Architecture:

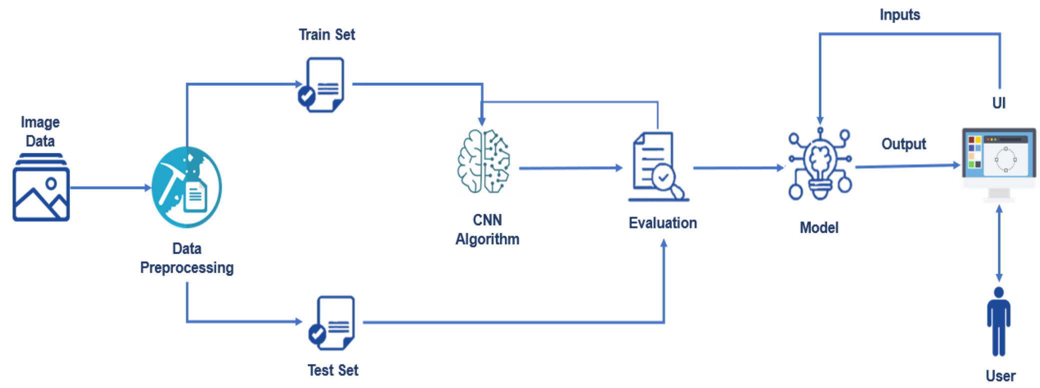


Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	Deep Learning Models	Image classification tasks, including breast cancer prediction.And analyze mammogram images and extract meaningful features for accurate predictions.	Convolutional Neural Networks (CNNs): TensorFlow, PyTorch, Keras
2.	Medical Imaging Data	Large datasets of labeled mammograms are required to develop robust prediction models. These datasets can be obtained from hospitals, medical research institutions, or publicly available repositories.	DICOM ,PACS
3.	Preprocessing Techniques:	Image enhancement, noise reduction, and normalization are applied to mammogram images before feeding them into deep learning models. Improve the quality and consistency of the input data, enhancing the model's performance	Image processing libraries: OpenCV, scikit-image, PIL (Python Imaging Library)

4.	Transfer Learning	Pre-trained deep learning models that have been trained on large-scale image datasets.	Pre-trained models: ResNet, VGG, Inception, DenseNet Frameworks: TensorFlow, PyTorch, Keras
5.	Data Augmentation:	Artificially increase the size of the training dataset & helps to introduce more diversity into the training data, reducing the risk of overfitting and improving generalization capabilities.	Image augmentation libraries: imgaug, Albumentations
6.	GPU Acceleration	Capable of parallel processing and provide the necessary computational power to handle large datasets and complex models efficiently.	INVIDIA CUDA toolkit for GPU acceleration
7.	Model Evaluation Metrics	To assess the performance of breast cancer prediction models.Also provide insights into the model's effectiveness and help compare different models.	Python libraries: scikit-learn, TensorFlow, PyTorch
8.	Deployment Frameworks	To integrate the trained model into a larger software system or create user-friendly applications for clinicians and radiologists.	TensorFlow, PyTorch, or Keras.

Table-2: Application Characteristics:

S.No	**Characteristics**	**Description**	**Technology**
1.	Accuracy	accurately predicting the presence or absence of breast cancer.	Deep learning models: TensorFlow, PyTorch, Keras

S.No	Characteristics	Description	Technology
			Evaluation metrics: scikit-learn, TensorFlow, PyTorch
2.	Speed and Efficiency	The prediction system should be efficient and capable of delivering results in a reasonable time frame.	Technology used
3.	Scalability	designed to handle large volumes of data and be scalable to accommodate increasing data sizes and user demand.	Distributed computing frameworks: Apache Spark, TensorFlow distributed training, PyTorch distributed training Cloud computing platforms: Amazon Web Services (AWS), Google Cloud Platform (GCP), Microsoft Azure Integration with Existing Workflows:
4.	Real-time or Near Real-time Performance	The system should provide predictions quickly enough to support timely decision-making and patient care.	High-performance computing (HPC) infrastructure Streaming frameworks: Apache Kafka, Apache Flink
6.	Generalization	Perform well on unseen data from different sources or	Data augmentation libraries:Albumentations

S.No	Characteristics	Description	Technology
		demographics.Techniques like data augmentation, transfer learning, and cross-validation can help improve generalization.	Transfer learning: TensorFlow, PyTorch, Keras
7.	Interpretability and Explainability	Making it difficult to understand how they arrive at their predictions.	Attention mechanisms: TensorFlow, PyTorch Saliency maps: TensorFlow, PyTorch Explainable AI libraries: SHAP (SHapley Additive exPlanations), LIME (Local Interpretable Model-agnostic Explanations)
8.	Integration with Existing Workflows	Involve interoperability with existing software systems and compliance with relevant standards, ensuring a smooth integration into the clinical environment.	Interoperability standards: DICOM (Digital Imaging and Communications in Medicine), HL7 (Health Level Seven International) Integration frameworks: Flask, Django, FastAPI for web service development EHR integration: FHIR (Fast Healthcare Interoperability Resources) standards

4.3. User Stories

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Team Member
Patient	Information Access	USN-1	Patients may have limited access to the system, but they can have access to relevant educational materials or information about breast cancer prevention, diagnosis, and treatment.	Patients have limited access to the system.	High	Praveena
	Data privacy	USN-2	Patients' privacy and confidentiality should be ensured, with the system adhering to relevant regulations, such as HIPAA, to protect their personal and medical information.	Patients' privacy and confidentiality is maintained .	High	
Healthcare Professionals	User Registration and Authentication	USN-3	Healthcare professionals should be able to create user accounts and authenticate themselves to access CancerVision.	To create user accounts & authenticate to access Cancervision .	Medium	Shree
	Image Upload and Processing	USN-4	Users should be able to upload breast cancer images, such as mammograms or	User to upload breast cancer images.	high	

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Team Member
			ultrasound scans, to the system for analysis.			
	Request Breast Cancer Prediction	USN-5	Healthcare professionals should have the ability to request breast cancer predictions for uploaded images.	To provide request for breast cancer predictions for uploaded images.	Medium	
	Prediction Results	USN-6	Users should receive accurate and timely prediction results, including the probability or likelihood of malignancy, tumor location, and other relevant information.	Users should receive accurate and timely prediction results.	High	
	Result Interpretation	USN-8	The system should provide tools or visualizations to assist healthcare professionals in interpreting the prediction results and making informed clinical decisions.	To assist healthcare and provide tools and visualizations for the system.	High	
Researchers and Developers	Model Training and Evaluation	USN-9	Researchers and developers should have the ability to train and evaluate deep learning	Developers should train and evaluate deep learning models	Medium	Kaviena

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Team Member
			models using breast cancer image datasets.			
	Model Integration	USN-10	Researchers and developers should be able to integrate trained models into the CancerVision system for deployment.	Researchers and developers integrate trained models into the CancerVision system.	Low	
	Algorithm Development	USN-11	Researchers and developers should have the flexibility to develop and enhance algorithms for image analysis and feature extraction.	Researchers have the flexibility to develop and enhance algorithms for image analysis and feature extraction.	Low	
	Testing and Validation	USN-12	Researchers and developers should be able to conduct testing and validation of the system to ensure accuracy and performance.	Researchers and developers conduct testing and validation ..	Medium	
System Administrator	User Management	USN-13	Administrators should have the capability to manage user accounts,	Administrators manage user	High	Taarakesh

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Team Member
			including creating, modifying, or deactivating accounts.	accounts, including creating, modifying, or deactivating accounts.		
	System Configuration	USN-14	Administrators should be able to configure system settings, such as deployment options, resource allocation, and security parameters.	Administrators configure system settings.	High	
	Performance Monitoring	USN-15	Administrators should have access to monitoring tools to track system performance, detect any issues, and ensure smooth operation.	Administrators monitor tools to track system performance, detect any issues.	High	

5. CODING & SOLUTIONING

5.1 Feature 1

1. **Data Preprocessing and Model Training:**

   - The breast cancer dataset is loaded using `sklearn.datasets.load_breast_cancer()`. This function retrieves a standard breast cancer dataset from the UCI Machine Learning Repository, which consists of clinical and mammographic features of breast masses.

   - The dataset is converted into a pandas DataFrame using `pd.DataFrame()` to facilitate data manipulation and analysis.

   - The column names of the DataFrame are set to `breast_cancer_dataset.feature_names`, which contains the names of the features in the breast cancer dataset.

   - A new column named 'label' is added to the DataFrame, which corresponds to the target variable or the presence of breast cancer. It is populated using `breast_cancer_dataset.target`.

   - The shape of the DataFrame is displayed using `data_frame.shape`, which provides the number of rows and columns in the dataset.

   - Information about the DataFrame, including the column data types and the presence of missing values, is obtained using `data_frame.info()` and `data_frame.isnull().sum()`, respectively.

   - Descriptive statistics of the DataFrame are computed using `data_frame.describe()`.

   - The count of each class label in the 'label' column is calculated using `data_frame['label'].value_counts()`.

5.2 Feature 2

2. **Model Evaluation and Prediction:**

   - The trained model is evaluated on the testing data using `model.evaluate(X_test_std, Y_test)`, which returns the loss and accuracy metrics.

   - Some sample data points are provided as `input_data` and preprocessed using the scaler (`scaler.transform(input_data_reshaped)`) to obtain standardized input features.

   - The preprocessed input data is used to make predictions using `model.predict(input_data_std)`. The predictions (`Y_pred`) are displayed, and the predicted labels (`Y_pred_labels`) are extracted.

5.3 Feature 3

3. **EDA and Data Visualization**:

- Another dataset is loaded from the file 'Cdata - Sheet1.csv' using `pd.read_csv('')`.

- The shape of the DataFrame is displayed using `breast_cancer_data.shape`.

- Information about the DataFrame, including the column data types and the presence of missing values, is obtained using `breast_cancer_data.info()` and `breast_cancer_data.isnull().sum()`, respectively.

- Descriptive statistics of the DataFrame are computed using `breast_cancer_data.describe()`.

- The count of each class label in the 'diagnosis' column is calculated using `breast_cancer_data['diagnosis'].value_counts()`.

- Label encoding is applied to convert the 'diagnosis' column into numeric form using `LabelEncoder`.

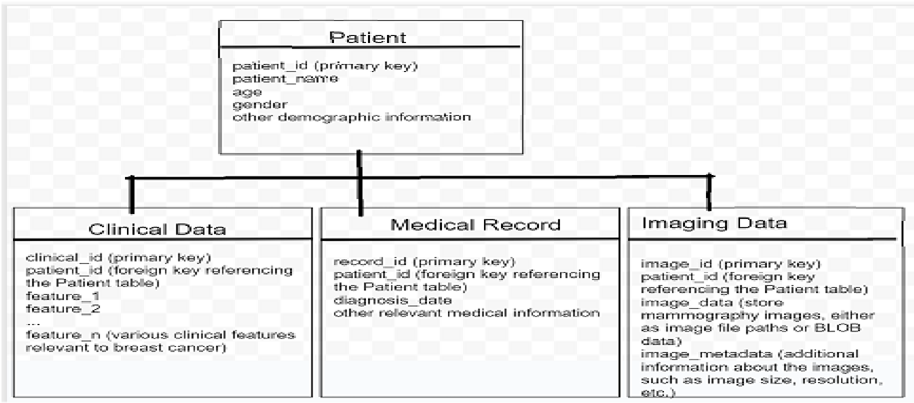
- The 'diagnosis' column is dropped from the DataFrame using `breast_cancer_data.drop(columns='diagnosis', axis=1, inplace=True)`.

- The count of each class label in the 'target' column (after label encoding) is calculated using `breast_cancer_data['target'].value_counts()`.

- Various visualizations are created using Seaborn and Matplotlib, including count plots (`sns.countplot()`), histograms (`sns.displot()`), scatter plots (`plt.scatter()`), and box plots (`breast_cancer_data.boxplot()`).

- A correlation matrix heatmap is generated using `sns.heatmap()` to visualize the pairwise correlations between the features in the DataFrame.

5.4 Database Schema

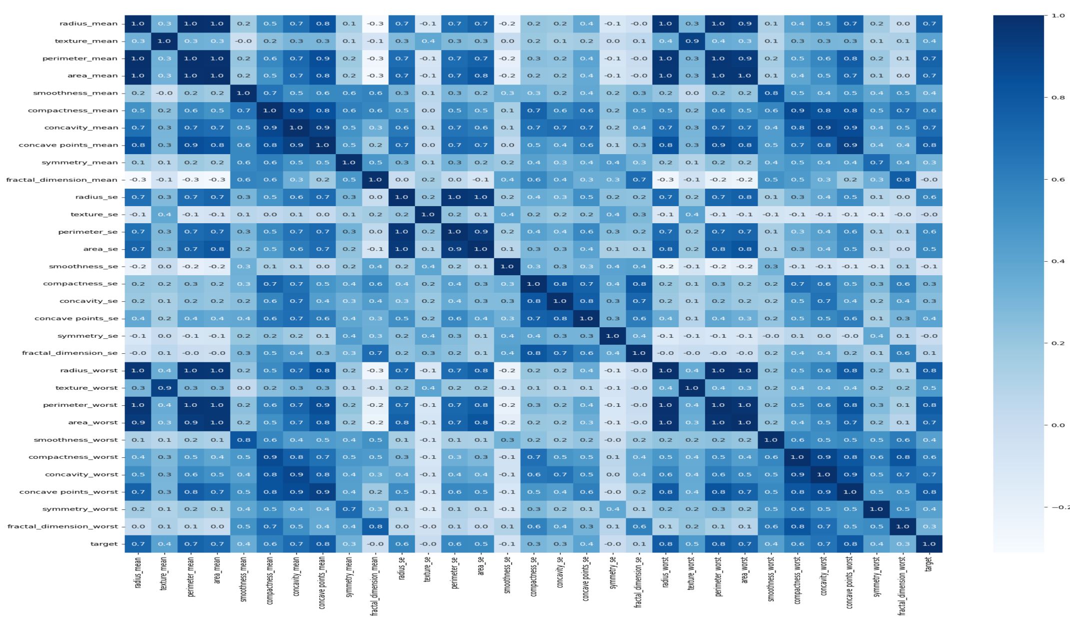


6. RESULTS

6.1 Performance Metrics

Model Performance Testing:

1) Model Summary:



2) Accuracy:

I. Training Accuracy - 0.9737

II. Validation Accuracy -0.978

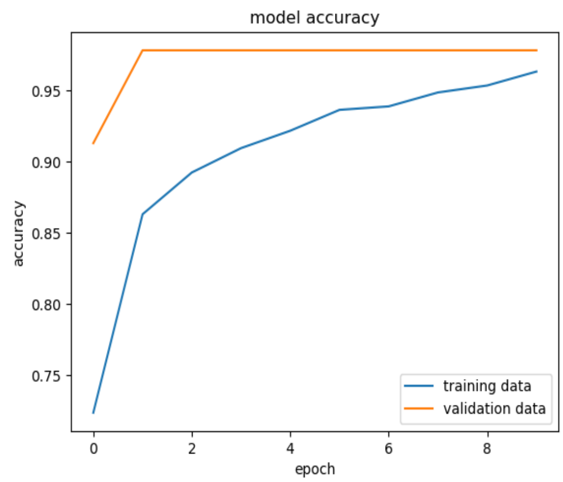
3

In [25]:
```
plt.plot(history.history['accuracy'])
plt.plot(history.history['val_accuracy'])

plt.title('model accuracy')
plt.ylabel('accuracy')
plt.xlabel('epoch')

plt.legend(['training data', 'validation data'], loc = 'lower right')
```

Out[25]: <matplotlib.legend.Legend at 0x16cfee7f1f0>



localhost:8888/notebooks/CANCER%20classification.ipynb

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3 (ipykernel)

In [27]:
```
loss, accuracy = model.evaluate(X_test_std, Y_test)
print(accuracy)
```

```
4/4 [==============================] - 0s 3ms/step - loss: 0.1273 - accuracy: 0.9737
0.9736841917037964
```

In [28]:
```
print(X_test_std.shape)
```

Hence we have created a model to predict whether the Tumour is malignant and benign which will be useful for helping the doctors and patients to classify the Type of Tumor.

7.ADVANTAGES AND DISADVANTAGES

Advantages of CancerVision: Advanced Breast Cancer Prediction with Deep Learning:

1.Early detection: Deep learning algorithms have the potential to analyze large amounts of medical data and identify patterns that may not be easily detectable by human physicians. CancerVision could enable earlier detection of breast cancer, which can significantly improve patient outcomes by allowing for timely treatment initiation.

2.Increased accuracy: Deep learning models can be trained on vast amounts of data, allowing them to recognize subtle patterns and features that might be missed by human observers. This could enhance the accuracy of breast cancer prediction, potentially reducing false negatives and false positives.

3.Efficiency and scalability: Once trained, deep learning models can process and analyze data quickly and efficiently. CancerVision could be implemented as an automated tool, capable of analyzing a large number of mammograms or other medical images within a short timeframe, thereby assisting healthcare providers and reducing the workload on radiologists.

Disadvantages of CancerVision: Advanced Breast Cancer Prediction with Deep Learning:

1.Data limitations and biases: Deep learning models require large amounts of high-quality data for training, and the availability of such data can be limited. In addition, if the training data contains biases (e.g., certain demographics being underrepresented), the model may inherit and amplify those biases, leading to disparities in accuracy across different populations.

2.Lack of interpretability: Deep learning models are often considered black boxes, as they make predictions based on complex computations that are challenging to interpret and explain. This lack of interpretability may limit the understanding of how and why CancerVision arrives at its predictions, making it difficult for healthcare professionals to trust and rely solely on the model's recommendations.

3.Regulatory and ethical considerations: Implementing CancerVision or any similar deep learning system in a clinical setting would require careful consideration of regulatory requirements and ethical implications. Privacy concerns, patient consent, data security, and liability are important factors that need to be addressed to ensure the responsible and safe deployment of such a system.

8. CONCLUSION

In conclusion, CancerVision, an advanced breast cancer prediction system utilizing deep learning, holds great promise for the future of breast cancer diagnosis and treatment. By harnessing the power of deep learning algorithms and analyzing comprehensive datasets, CancerVision has the potential to significantly improve the accuracy of breast cancer prediction, enabling timely interventions and personalized treatment strategies.

The future scope for CancerVision encompasses various areas of development. Enhancing accuracy through algorithm refinement, optimizing model architecture, and incorporating additional data sources can further enhance its predictive capabilities. Additionally, expanding CancerVision's scope to include early detection of breast cancer can lead to improved patient outcomes by enabling early intervention and treatment.

Integration with medical imaging technologies, such as mammography or MRI, can provide a more comprehensive assessment of breast cancer risk. By combining deep learning-based predictions with imaging data, healthcare providers can obtain a more accurate and holistic evaluation of a patient's breast health.

9.FUTURE SCOPE

The future scope for CancerVision, an advanced breast cancer prediction system using deep learning, is promising and encompasses several areas of potential development and improvement.

1.Enhanced Accuracy: Continuously improving the accuracy of CancerVision's breast cancer prediction capabilities should be a primary focus. Researchers can explore ways to refine the deep learning algorithms, optimize the model architecture, and incorporate additional relevant data sources to enhance the system's ability to accurately predict advanced breast cancer.

2.Early Detection: Expanding CancerVision's capabilities to include early detection of breast cancer can significantly improve patient outcomes. By training the deep learning model on comprehensive datasets that include information from various stages of breast cancer development, it may be possible to identify subtle patterns or indicators that precede advanced stages, enabling early intervention and treatment.

3.Integration with Medical Imaging: Integrating CancerVision with medical imaging technologies, such as mammography, magnetic resonance imaging (MRI), or ultrasound, can provide a more comprehensive assessment of breast cancer risk. By combining deep learning-based prediction models with imaging data, doctors and radiologists can obtain a more accurate and holistic evaluation of a patient's breast health.

10. APPENDIX

Source Code :

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import sklearn.datasets
from sklearn.model_selection import train_test_split
breast_cancer_dataset = sklearn.datasets.load_breast_cancer()
print(breast_cancer_dataset)
data_frame = pd.DataFrame(breast_cancer_dataset.data, columns =
breast_cancer_dataset.feature_names)
data_frame.head()
data_frame['label'] = breast_cancer_dataset.target
data_frame.tail()
data_frame.shape
data_frame.info()
data_frame.isnull().sum()
data_frame.describe()
data_frame['label'].value_counts()
data_frame.groupby('label').mean()
X = data_frame.drop(columns='label', axis=1)
Y = data_frame['label']
print(X)
print(Y)
X_train, X_test, Y_train, Y_test = train_test_split(X, Y,
test_size=0.2, random_state=2)
```

```
print(X.shape, X_train.shape, X_test.shape)

from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()
X_train_std = scaler.fit_transform(X_train)
X_test_std = scaler.transform(X_test)
import tensorflow as tf
tf.random.set_seed(3)
from tensorflow import keras

model = keras.Sequential([
                          keras.layers.Flatten(input_shape=(30,)),
                          keras.layers.Dense(20,
activation='relu'),
                          keras.layers.Dense(2,
activation='sigmoid')
])


model.compile(optimizer='adam',
              loss='sparse_categorical_crossentropy',
              metrics=['accuracy'])


history = model.fit(X_train_std, Y_train, validation_split=0.1,
epochs=10)


plt.plot(history.history['accuracy'])
plt.plot(history.history['val_accuracy'])


plt.title('model accuracy')
plt.ylabel('accuracy')
```

```
plt.xlabel('epoch')

plt.legend(['training data', 'validation data'], loc = 'lower right')

plt.plot(history.history['loss'])
plt.plot(history.history['val_loss'])

plt.title('model loss')
plt.ylabel('loss')
plt.xlabel('epoch')

plt.legend(['training data', 'validation data'], loc = 'upper right')

loss, accuracy = model.evaluate(X_test_std, Y_test)
print(accuracy)

print(X_test_std.shape)
print(X_test_std[0])

Y_pred = model.predict(X_test_std)

print(Y_pred.shape)
print(Y_pred[0])

print(X_test_std)
```

```
print(Y_pred)

my_list = [0.25, 0.56]

index_of_max_value = np.argmax(my_list)
print(my_list)
print(index_of_max_value)



Y_pred_labels = [np.argmax(i) for i in Y_pred]
print(Y_pred_labels)

input_data =
(11.76,21.6,74.72,427.9,0.08637,0.04966,0.01657,0.01115,0.1495,0.05888,0.4062,1.21,2.635,28.47,0.005857,0.009758,0.01168,0.007445,0.02406,0.001769,12.98,25.72,82.98,516.5,0.1085,0.08615,0.05523,0.03715,0.2433,0.06563)

input_data_as_numpy_array = np.asarray(input_data)

input_data_reshaped = input_data_as_numpy_array.reshape(1,-1)

input_data_std = scaler.transform(input_data_reshaped)

prediction = model.predict(input_data_std)
print(prediction)
```

```
prediction_label = [np.argmax(prediction)]
print(prediction_label)

if(prediction_label[0] == 0):
  print('The tumor is Malignant')

else:
  print('The tumor is Benign')



#"EDA and Data visualizatio of the breast cancer classification"

# Data visualization
import seaborn as sns
from sklearn.preprocessing import LabelEncoder





breast_cancer_data = pd.read_csv('C:/Users/Kaviena
Sharon/Downloads/Cdata - Sheet1.csv')

breast_cancer_data.head()

breast_cancer_data.drop(columns='Unnamed: 32', axis = 1,
inplace=True)
```

```
breast_cancer_data.head()

breast_cancer_data.shape

breast_cancer_data.info()

breast_cancer_data.drop(columns='id', axis=1, inplace=True)

breast_cancer_data.isnull().sum()

breast_cancer_data.describe()

breast_cancer_data['diagnosis'].value_counts()

label_encode = LabelEncoder()

labels =
label_encode.fit_transform(breast_cancer_data['diagnosis'])

breast_cancer_data['target'] = labels

breast_cancer_data.drop(columns='diagnosis', axis=1, inplace=True)
```

```
breast_cancer_data.head()



breast_cancer_data['target'].value_counts()



sns.countplot(x='target', data=breast_cancer_data)




breast_cancer_data.groupby('target').mean()


sns.countplot(x='target', data=breast_cancer_data)


for column in breast_cancer_data:
  print(column)


for column in breast_cancer_data:
  sns.displot(x=column, data=breast_cancer_data)


sns.distplot(x=breast_cancer_data.radius_mean)


first_column = breast_cancer_data.iloc[:, 0]
second_column = breast_cancer_data.iloc[:, 1]
print(first_column)
print('-----')
```

```
print(second_column)

plt.scatter(x=first_column, y=second_column)
for column in breast_cancer_data:
    plt.figure()
    breast_cancer_data.boxplot([column])


correlation_matrix = breast_cancer_data.corr()

plt.figure(figsize=(20,20))
sns.heatmap(correlation_matrix, cbar=True, fmt='.1f', annot=True,
cmap='Blues')
plt.savefig('Correlation Heat map')
```

Github:

https://github.com/naanmudhalvan-SI/PBL-NT-GP-14480-1683893783

Project Video Demo Link:
Click here