Using Machine Learning and Computer Vision to Detect Breast Cancer

Group 3:
Ahmed Ahmed
Mike Moll
Lisa Vo

Table of Contents

Background

Recent breast cancer trends

- About 297,790 new cases in 2023
- About 43,700 women will die in 2023
- 2nd most common cancer among women (behind skin cancer)
- Median age: 62
- Likelihood: 13% or 1 in 8 women

Our Goal

- To train a classifier using support vector machines (SVMs) to predict that a patient has breast cancer based on tabular patient data
- To train another classifier using convolutional neural networks (CNNs) to predict that a patient has breast cancer based on patient mammograms
- To compare the accuracy between SVMs and CNNs in predicting breast cancer in patients

Our Dataset

- Our dataset comes from CBIS-DDSM (Curated Breast Imaging Subset of DDSM) WHICH is an updated and standardized version of the Digital Database for Screening Mammography (DDSM).

Why did we choose to compare these two models?

- Our experiment is a reflection of what we've learned in this course
- We are exploring a mix of machine learning, deep learning, and computer vision concepts
- We are comparing two different approaches to classification using two different types of datasets (image and tabular data)

Support Vector Machine

Tabular Data Pre-Processing

- Removed unneeded columns: patient_id, abnormality_type, image file path, cropped image file path, and ROI mask file path
- Replace NaN with mode of corresponding feature

Features

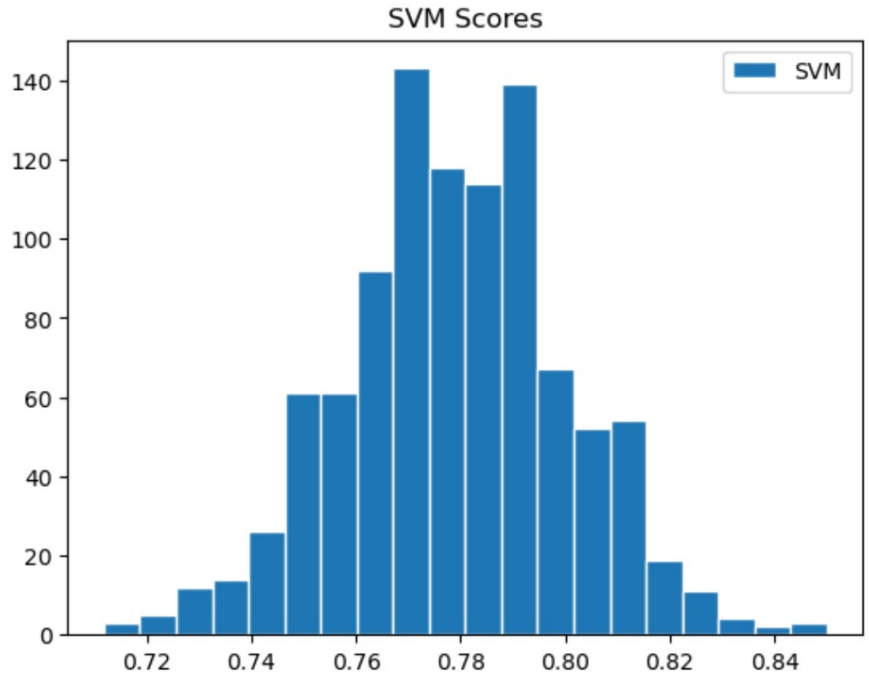
- **Target variable**
  - pathology: “BENIGN”, “MALIGNANT”, “BENIGN_WITHOUT_CALLBACK”
- **Explanatory variables:**
  - breast density, left or right breast, image view, abnormality id, mass shape, mass margins, assessment, and subtlety

Model Training

1. x_train, x_test, y_train, y_test was created from scikit-learn’s train_test_split with test size of 0.20
2. Support Vector Classification (SVC) object used to predict the data points
3. Accuracy scores were calculated from the predictions and y_test data
4. Predictions and accuracy scores calculated 1000 times for score distribution

Discussion & Results (1/2)

Distribution Histogram



Discussion & Results (2/2)

1. Mean accuracy score = 0.78
2. Standard deviation = 0.02
3. Median = 0.78
4. Minimum accuracy score reached = 0.71
5. Maximum accuracy score reached = 0.84

Convolutional Neural Network

Image Pre-Processing

1. Conversion of DICOM images to jpg
2. Three directories of labeled images (Training, Validation, and Test)
3. Each directory contains 3 subfolders of labelled jpg images (Benign, malignant, Benign call back)
4. Image size = 128/ 128
5. Class labels = ['BENIGN', 'MALIGNANT', 'BENIGN_WITHOUT_CALLBACK']

Layers

1- Input Image:

- Dimensions: 128x128 pixels
- Channels: 3 (RGB)

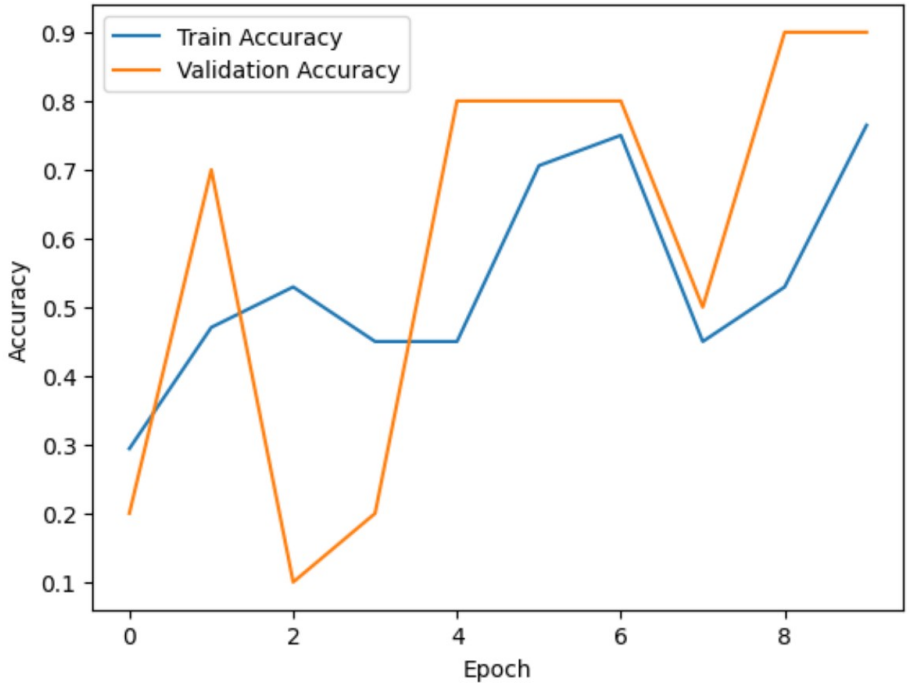
2- Layers:

- Convolutional Layer: 32 filters of size 3x3 with ReLU activation.
- Max Pooling Layer: 2x2 pooling size.
- Convolutional Layer: 64 filters of size 3x3 with ReLU activation.

Layers

- Max Pooling Layer: 2x2 pooling size.
- Flattening Layer: Convert 2D feature maps to 1D feature vector.
- Dense Layer: 256 neurons with ReLU activation.
- Dropout Layer: 50% dropout rate to prevent overfitting.
- Output Layer: 3 neurons (for three classes) with Softmax activation.

Model Training



Model Training

The model could not perform the prediction process. This could be related to incorrect loading or labeling of the images from the directory. There's a possibility that the model consistently predicts the same class for all images due to poor training or issues with the training data.

Discussion & Results

- Need to complete this section

Future Research & Implications

Benefits of Our Experiment

- Deeper understanding of computer vision and machine learning concepts
- Applied concepts learned, at a deeper level
- Expanded technical skills

Potential Improvements in Experiment

- Add content here

Thank you