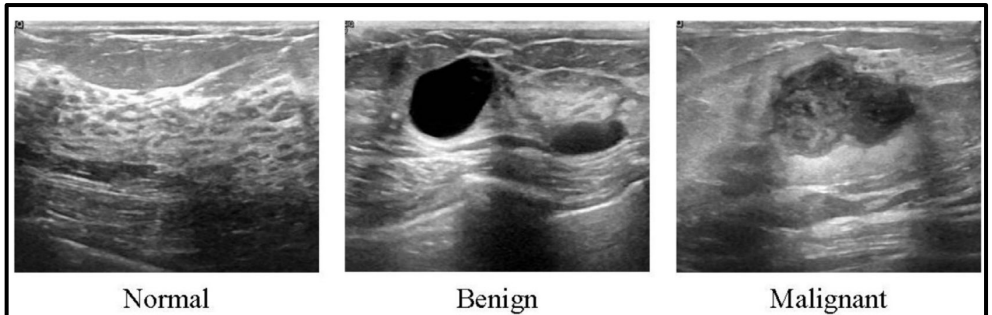
Breast Cancer Detection

Kishan Shah, Sruthi Machina

Background Information

- **Breast Ultrasound Images (BUSI)**
  - Kaggle data: <https://www.kaggle.com/datasets/aryashah2k/breast-ultrasound-images-dataset>
  - Data source location: Baheya Hospital for Early Detection & Treatment of Women's Cancer, Cairo, Egypt.
  - Collected in 2018 from 600 female patients, ages between 25 and 75 years old
  - Published by Al-Dhabyani et. al. from Cairo University, Egypt

- **BUSI dataset characteristics:**
  - 780 images, average image size of 500 x 500 pixels in PNG format
  - Three classes, normal (133), benign (437), and malignant (210)



Problem Statement

- **Differentiate between benign and malignant tumors in BUSI**
  - Normal (no tumor) cases are trivial to identify
  - Important to reduce both false positives and false negatives
  - Visual inspection:
    - Both benign and malignant tumors come in all shapes and sizes
    - Malignant tumors tend to have rough edges, with lower opacity
- **Our approach:**
  - Analyze images for transforms that make it easy to differentiate (benign and malignant)
    - Frequency, histogram, edge and corner detection, and feature descriptors
  - Use both traditional and deep learning techniques:
    - Logistic regression, Bag of Visual Words (BoVW), and CNN
  - Experiment with promising features and hyperparameter tuning
  - Apply clustering and dimensionality reduction to aid (K-Means and PCA) in classification

Pre-processing Steps

Problem: Image sizes are not uniform

- Average image size of **500 x 500**. Most algorithms require fixed feature sizes.

Solution:

- Resize to **512 x 512**. Some distortion, but not significant. Better than cropping.
- Downsampling to **256, 128, 64**, and **32** for testing

Problem: Dataset is not balanced

- Positive (malignant) images: **210** Negative (benign) images: **437**

Solution:

- Use image augmentation for positive cases.
- Mirror the positive images to balance the dataset (**210 → 420**)
- **437** benign cases, **420** malignant cases, Total **857**

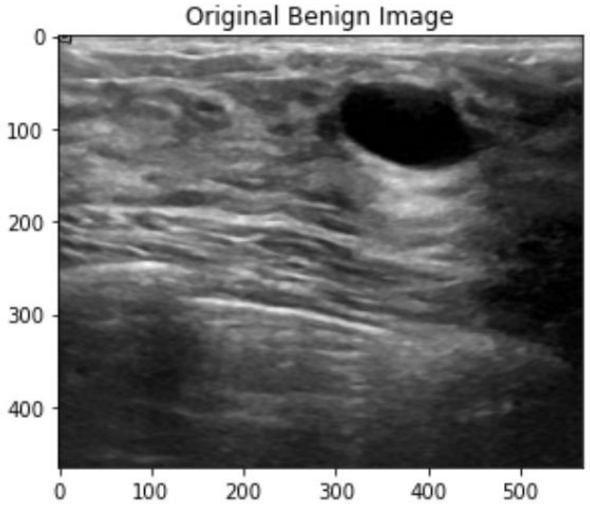
Feature and Filter Analysis of images

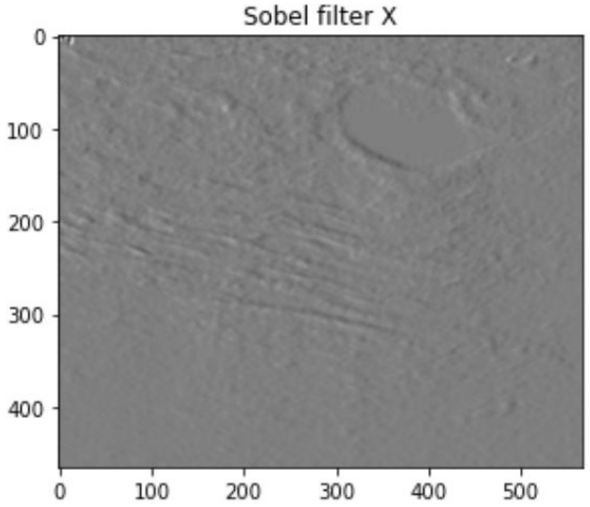
Notebook: https://github.com/shahkb/project_BUSI/blob/main/image_analysis.ipynb

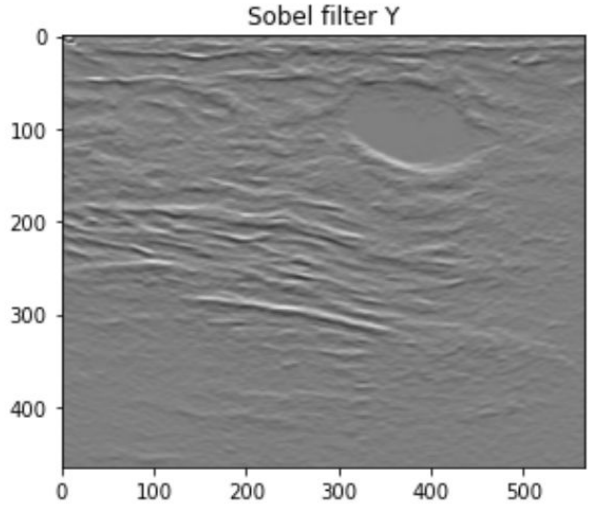
Feature / Filter	Potential to differentiate
Histogram, Fourier signal	No
Pre-filter: Histogram equalization	No
Pre-filter: Gaussian	Yes
Edge feature: Canny	No
Edge feature: Sobel	Yes
Corner detection: Harris and Shi-Tomasi	No
Feature descriptors (SIFT and ORB)	Yes

Custom Feature – Sobel Filter

Tried: Edge Detection (Prewitt Edge Detection - horizontal and vertical, Canny filter - different sigmas)

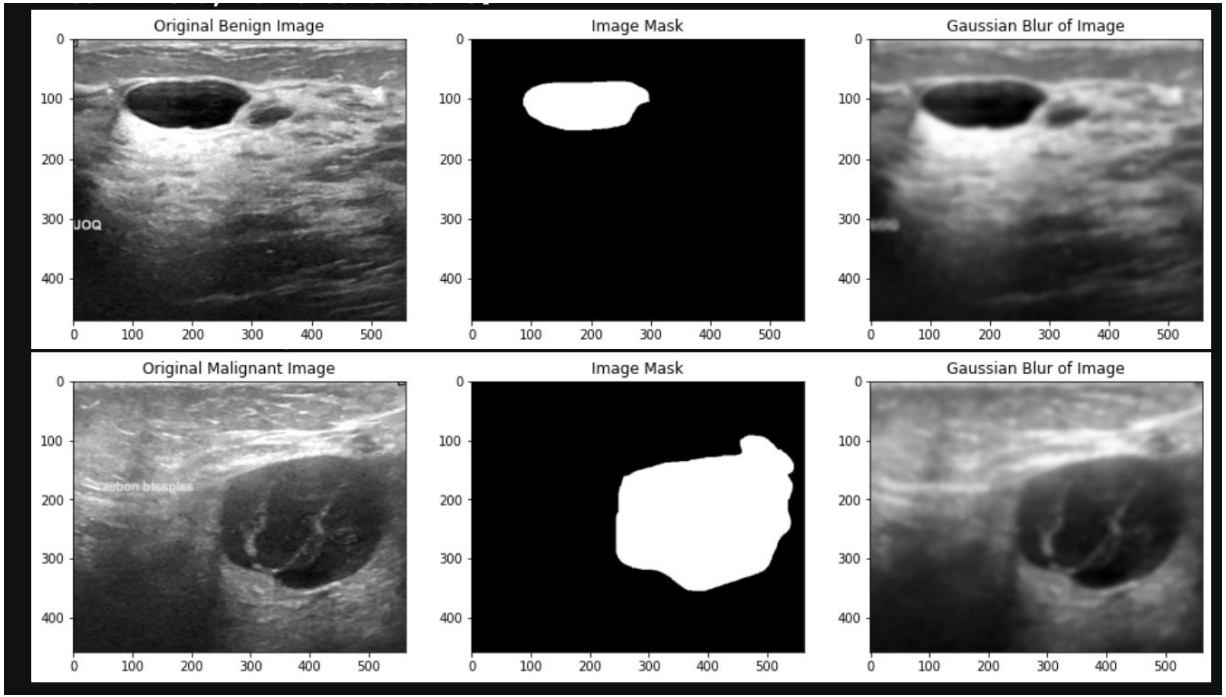






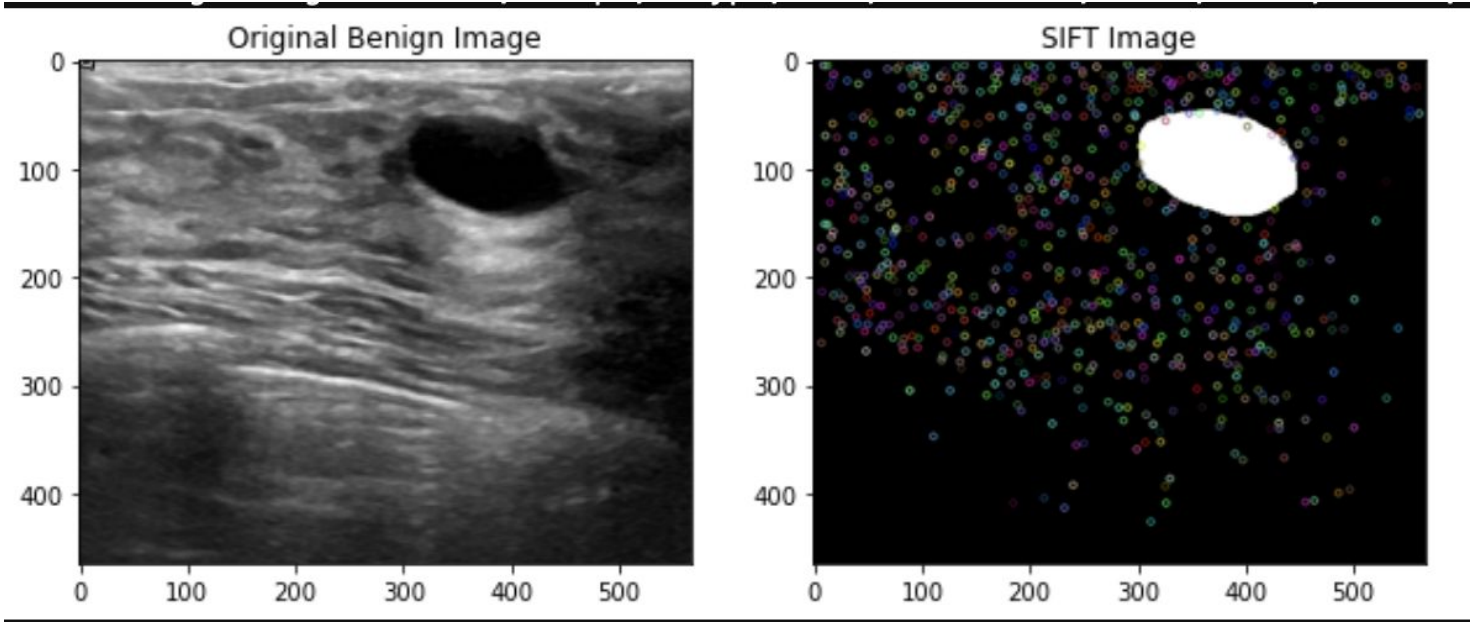
Custom Feature – Gaussian Filter

Tried: Low Pass Filtering - Laplacian , mean filter



Custom Feature – SIFT

Tried: Corner Detection (Harris Corners, Shi-Tomasi, ORB - Oriented Fast and Rotational Brief)



PCA & t-SNE Analysis

Image resolution (512 x 512)

- Feature vector size (**m**) 262,144

Number of samples 857

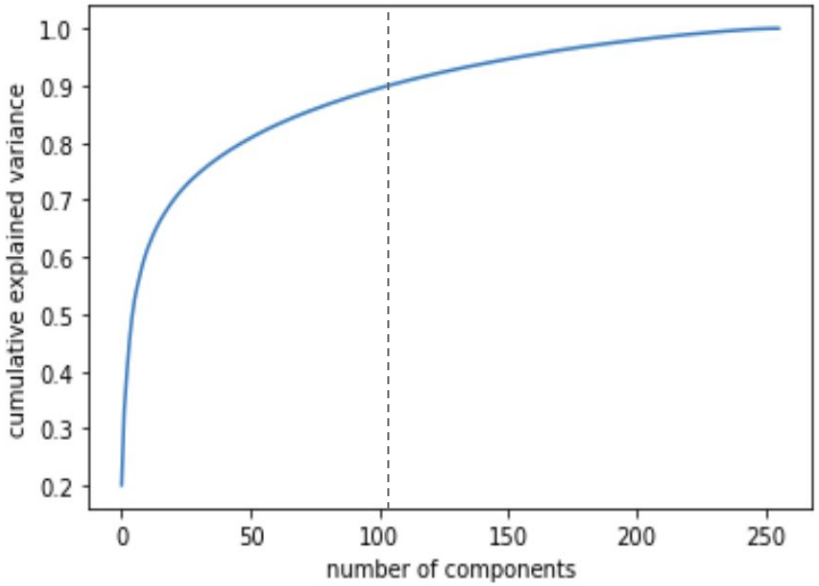
- Sample size (**n**) includes benign and malignant images
- **n << m**

Data Properties

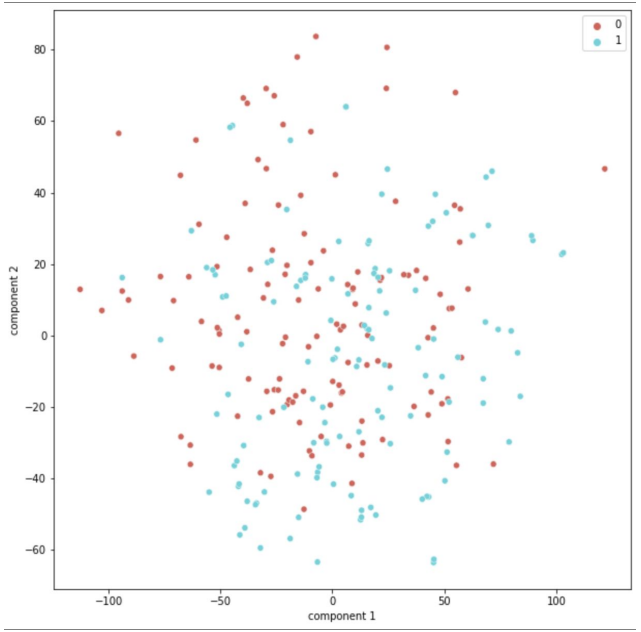
- Image resolution (**256 x 256**) m = 65,536
- Random sample (balanced) n = 256

PCA Analysis

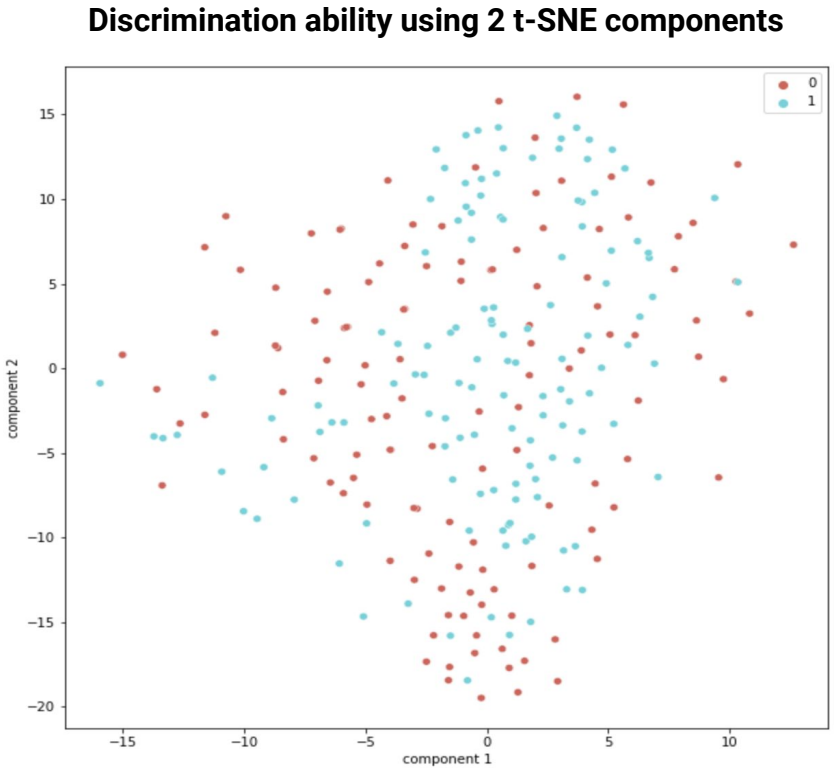
Num of PCA components required to explain % of variation



Discrimination ability using 2 PCA components



t-SNE Analysis



Classification Pipeline

1. **Balance the dataset**
2. **Resize images to a uniform resolution**
3. **Generate feature (Raw pixel, Gaussian & Sobel filters, SIFT, ORB)**
4. Dimensionality reduction (PCA), Clustering (K-Means)
5. **Split dataset into Test-Train (30% -70%), Scale the data (Standard scaler)**
6. **Specify a learning model and choose hyper-parameters**
7. **Train the model**
8. **Analyze result (on test dataset)**
9. **Tune the model (using Grid Search)**

Traditional Classification

Logistic Regression

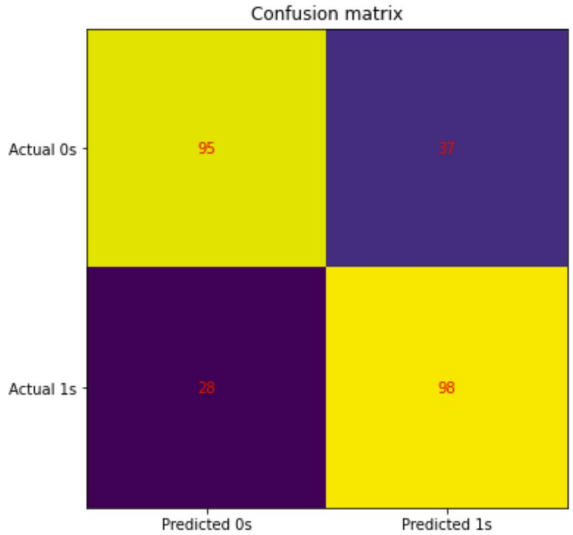
1. **Logistic regression features:**
    a. Raw pixels
    b. Gaussian and Sobel filters
    c. PCA components as features

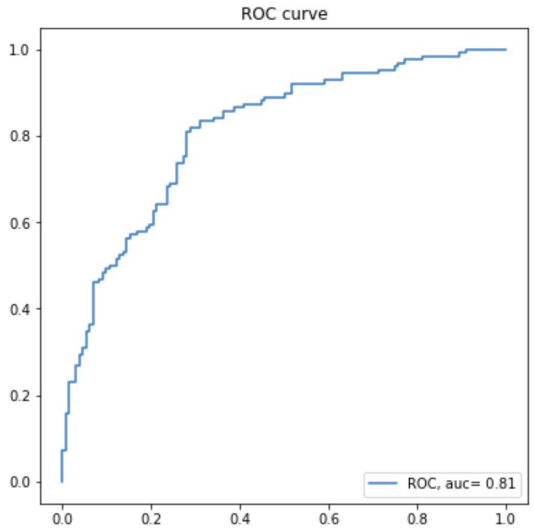
2. **Model tuning**
    a. Type of regularization ('l1', 'l2')
    b. Strength of regularization (C = 0.001, 0.005, 0.01, 0.05, 0.1, 0.5, 1.0, 10)
    c. Type of solver ('liblinear')

Baseline result for Logistic Regression

Baseline (features: Raw pixel, C= 1.0)

Accuracy= 0.75, f1-score= 0.75





Fine tuning Logistic Regression

Feature	Norm	C	Accuracy	f1-score	CPU sec
Raw pixels	L1	0.1	0.77	0.76	504
Sobel filter	L2	0.01	0.64	0.65	400
Gaussian filter	L1	1.0	0.78	0.78	726
PCA components	L1	0.005	0.73	0.73	19

PCA variance= 90%
Dimensionality reduction from (599, 65536) to (599, 178)
Speedup x20 to x40

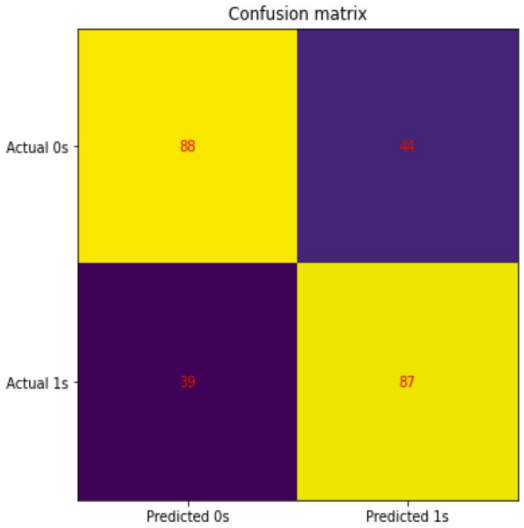
Bag of Visual Words

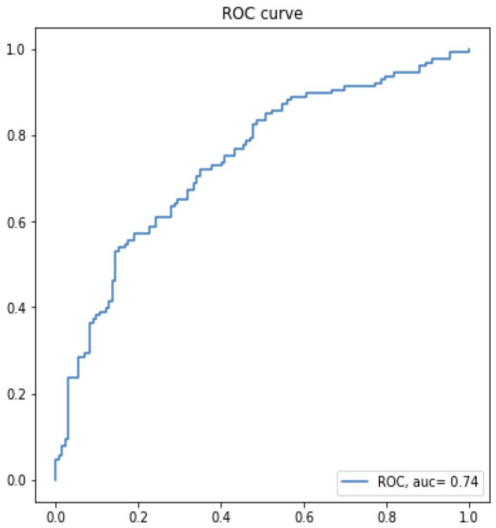
1. **Steps**
    a. Extract feature descriptors (**SIFT and ORB**)
    b. Cluster features using **K-Means**, clusters become the visual words
    c. Generate feature data
        i. Histogram: how many of each visual words occur in an image
    d. Apply learning model (**Logistic regression, SVM**)
2. **Model tuning**
    a. Number of **features** extracted by SIFT and ORB
    b. Number of **clusters** in K-Means (vocabulary size)
    c. Learning model parameters (**C** - regularization strength)

Baseline result for Bag of Visual Words

Baseline (ORB, nfeatures= 400, nclusters= 200, Logistic Regression, C= 1.0, L2)

Accuracy= 0.69, f1-score= 0.68





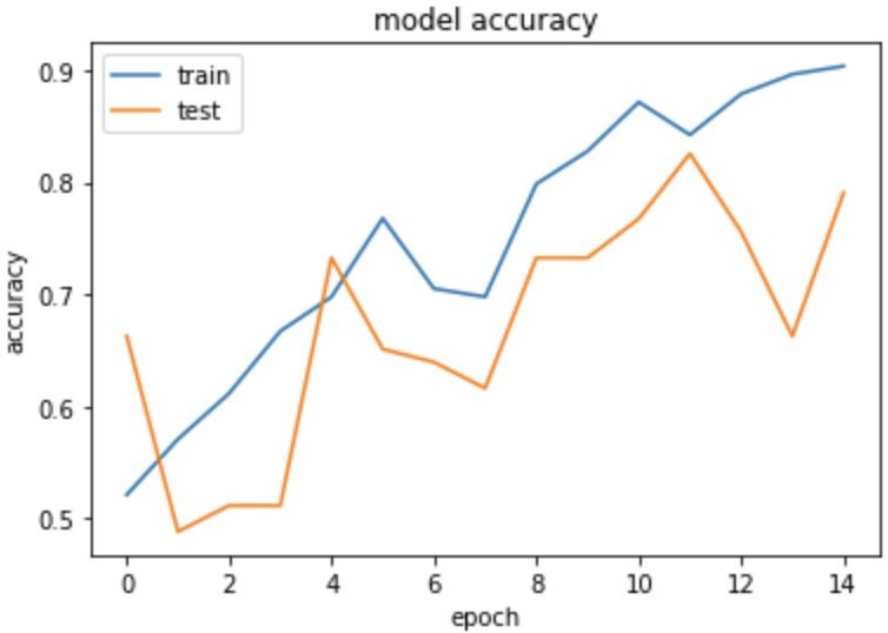
Fine tuning Bag of Visual Words

Feature	nfeatures	nclusters	model	C	Accuracy	f1-score
ORB	400	200	Logistic R	0.001	0.72	0.71
ORB	800	400	Logistic R	0.01	0.74	0.74
ORB	800	400	SVM	0.001	0.74	0.73
SIFT	400	200	Logistic R	0.01	0.73	0.71
SIFT	800	400	Logistic R	0.005	0.74	0.74
SIFT	800	400	SVM	0.001	0.76	0.76

Deep Learning

Model: CNN

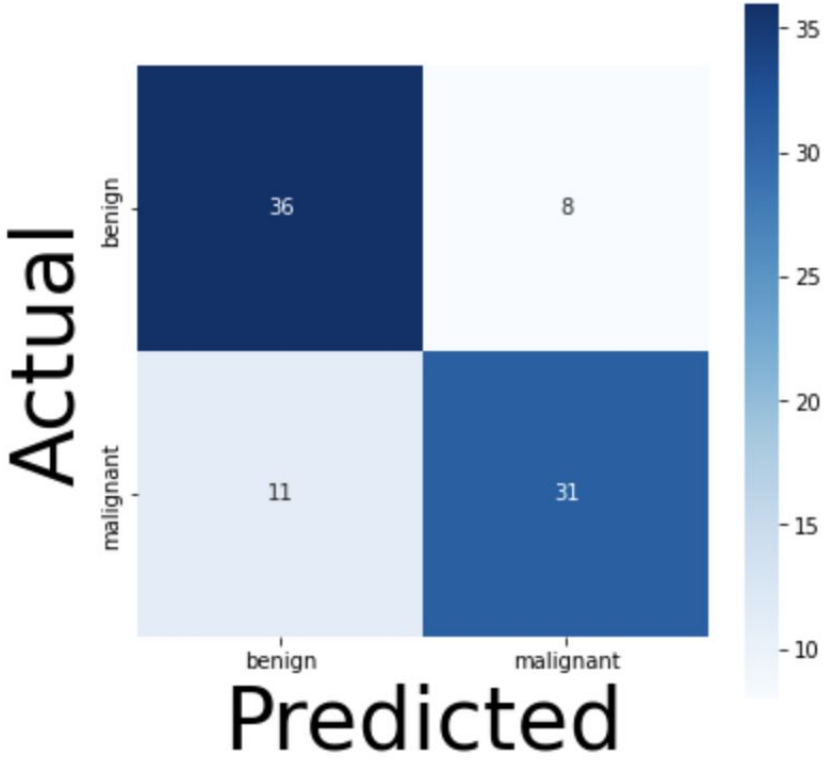
Model Accuracy Over Training Iterations



Tuned Parameter

- Batch size (16, 32, 64)
- Optimization (adam, adagrad)
- Learning rate (.001, 01)
- Epochs (10, 15, 20)

Model: CNN


Actual
benign
malignant
36
8
11
31
benign
malignant
Predicted
35
30
25
20
15
10

Summary and Limitations

- The types of distinctions within each category of tumor are extremely varied and not always unique to each category (i.e. size or location on the picture)
- Logistic regression with pixels as features kept overfitting on the training data, regardless of using regularization terms.
- Logistic regression with PCA shows much lower overfitting, and huge speedup
- Bag of Visual Words shows lower overfitting. But more fine tuning needed
- Most custom features worked very poorly to identify the tumor modules – even the ones we ended up choosing which were the best of all of the filters and learned features we tried

References

- Al-Dhabyani W, Gomaa M, Khaled H, Fahmy A. Dataset of breast ultrasound images. Data in Brief. 2020 Feb;28:104863. DOI: 10.1016/j.dib.2019.104863.
- Kaggle dataset: https://www.kaggle.com/datasets/aryashah2k/breast-ultrasound-images-dataset
- GitHub repository for notebooks
  - https://github.com/shahkb/project_BUSI