Breast Cancer Detection

RAVINA LAD

28th April 2022

Overview

Breast cancer is the second leading cause of death in women. Breast cancer mortality is reduced by early identification and treatment.

Implementation of this project can help in the following ways:

1. Help physicians for early detection to maximize patients' survival rate.
2. Minimize the number of "untrained eyes" that is wrong interpretations and increase the accuracy of screening.
3. Prevent late treatments as well as unnecessary treatments in case of false positives.
4. One would be able to overcome the dependency of pathologist in the places where no experts are available.

Dataset

The dataset to be used in this project is put together by Mendeley Data. It is free to use and open source. The dataset contains mammography with benign and malignant masses.

1. INbreast

    Total Images : 7632 Types of Breast Masses : 106

2. MIAS (Mammographic Imaging Analysis society)

    Total Images : 3816 Types of Breast Masses : 53

Training and Validation set contains : 11448

Test set contains : 2349

Image Size : 227 * 227.

Approach

- Image Processing :
  1. Image transformations: reshape, Rotation, sheering,
  2. Data Augmentation

- Feature Extraction :
  1. Patterns and textures extracted using combination of convolution and max pooling layers.
  2. CNN filters and their weights, are features that are used at the time of testing for model evaluation.

- Classification :
  1. Places an image into the respective class (benign or malignant)
  2. Classified using a fully connected layer using an activation function such as Softmax.

- Basic CNN : 86.4 %
  Adding more layers (64 channels + 3 x 3 kernel) : 90 %
  Architecture + Image Augmentation : 91 %
  Final Accuracy (32@5X5 + 64@5x5 + 128@5x5):

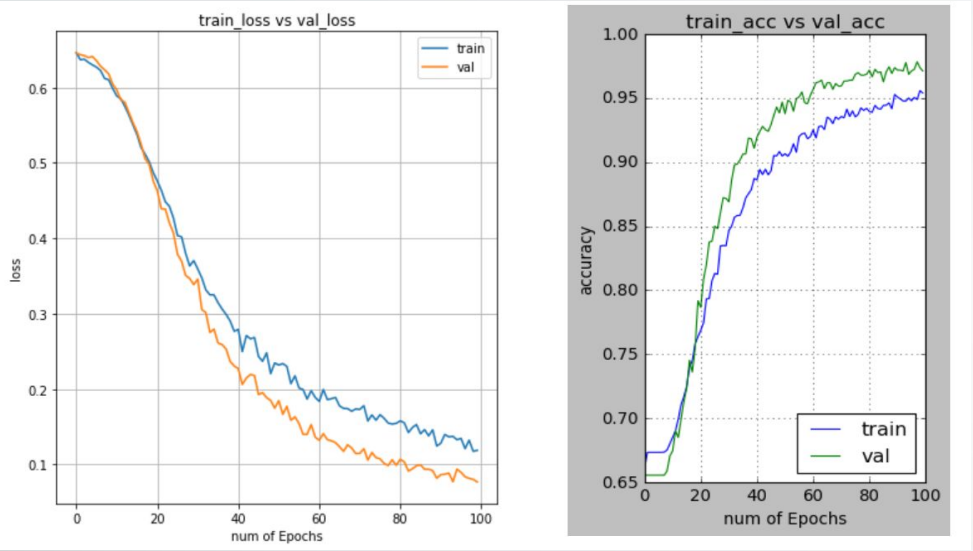
- 1. MIAS : 92.02 %

- 2. INbreast : 97.45 %

CNN Model Architecture

```
# define the larger model
def breastCancer_model():
    # create model
    model = Sequential()
    model.add(Conv2D(32, (3, 3), padding="same",input_shape=(92,140,3), activation='relu'))
    #model.add(Conv2D(32, (3, 3), activation='relu',padding = 'same'))
    model.add(MaxPooling2D(pool_size=(2, 2)))
    model.add(Conv2D(32, (3, 3), activation='relu',padding = 'same'))
    #model.add(Conv2D(64, (3, 3), activation='relu',padding = 'same'))
    model.add(MaxPooling2D(pool_size=(2, 2)))
    model.add(Conv2D(64, (3, 3), activation='relu',padding = 'same'))
    #model.add(Conv2D(128, (3, 3), activation='relu',padding = 'same'))
    model.add(MaxPooling2D(pool_size=(2, 2)))
    model.add(Dropout(0.5))
    model.add(Flatten())
    model.add(Dropout(0.5))
    model.add(Dense(64, activation='relu'))
    model.add(Dropout(0.5))
    model.add(Dense(64, activation='relu'))
    model.add(Dropout(0.5))
    #model.add(Dense(50, activation='relu'))
    #model.add(Dropout(0.2))
    model.add(Dense(num_classes, activation='softmax'))
    # Compile model
    model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])
    return model
```

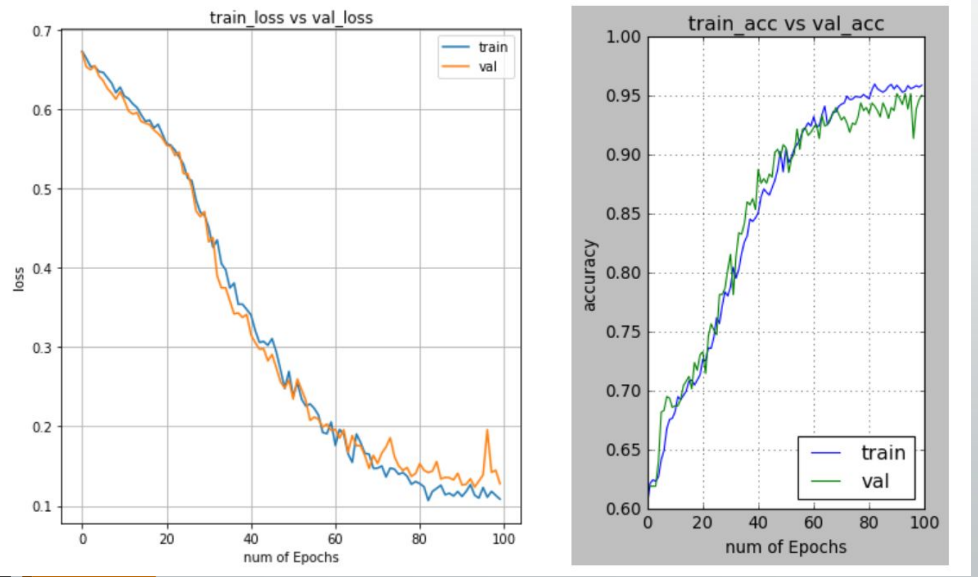
Performance Graphs



- Training loss and validation loss.

Loss start with a high value and decrease while training proceeds.

Difference in saturation levels of Training Loss and Validation Loss is 0.2, within the permissible range for a network to avoid underfitting or overfitting.



- Training accuracy and validation accuracy.

Accuracy starts to increase with the number of epochs, and ultimately saturates.

No underfitting and overfitting, as validation accuracy and training accuracy curves are similar in distribution.

Classification Report

	precision	recall	f1-score	support
0	0.92	0.95	0.94	467
1	0.92	0.87	0.89	285
accuracy			0.92	752
macro avg	0.92	0.91	0.91	752
weighted avg	0.92	0.92	0.92	752

- **Precision :**

  A probabilistic measure to determine whether a positive case, defined on our terms, actually belongs to positive case.

	precision	recall	f1-score	support
0	0.95	0.98	0.96	526
1	0.99	0.97	0.98	1001
accuracy			0.97	1527
macro avg	0.97	0.98	0.97	1527
weighted avg	0.97	0.97	0.97	1527

- **Recall and F1 score :**

  A probabilistic measure to determine if an actual positive case is correctly classified with the positive class.

  F1 score is calculated as the geometric mean between precision and recall.

Conclusion & Future scope

	A	B	C	D
1	Year	Method Used	Accuracy(%)	Error Rate
2	2017	K-Nearest Neighbor [12]	83 to 86	19.28
3	2019	Pre-Trained Networks [10]	90 to 97	4.74
4	2017	Feature Extracted Using CNN	83 to 90	4.28
5	2018	Deep Convolution Neural Network [11]	91.54	8.54

- For comparison, we have compared our result (92.02% and 97.45% validation accuracy from test set)with several published studies

1. Results are insensitive to the resolution of images.
2. Scope to implement auto-encoders instead of manually reducing image size.
3. It can compress data without losing the prominent features.
4. Classified breast cancer tissues into two classes benign and malignant with :
    - 4.1 Accuracy of 92.02 % for MIAS
    - 4.2 Accuracy of 97.45% for INbreast datasets.

THANK YOU !