# *Breast Cancer Classification with*

# *Deep Learning*

***Introduction:***

To build a breast cancer detection an IDC dataset that can correct describe a histology image as malign. In this project of the python, we’ll build a classifier to train on 80% of a breast cancer image dataset. Of this, we’ll keep 10% of the data for validation. Using Keras, we’ll define a [***CNN (Convolutional Neural Network)***](https://data-flair.training/blogs/convolutional-neural-networks-tutorial/), and train it on our images. and then we will find a confusion matrix to analyze the performance of the model.

IDC is the form of Invasive Ductal Carcinoma cancer that found in a milk tube and occupy the fatty breast tissue outside the tube it is the most common form of breast cancer forming 80% of all breast cancer diagnoses. And histology is a structure of tissues.  Breast Cancer is the most common type of cancer in woman worldwide accounting for 20% of all cases...

In 2012, it is found that 1.68 million new cases are recorded and 522,000 deaths.

The main problems are that women often abandon the indication, which causes more bad effects on them thus lowering the survival chances. In modern countries, the survival rate is although high but in the area of developing countries where the 5% survival rates are poor. In India, there are about one million cases every year and survival percentage is 10%. Therefore, it is very important to detect the signs as early as possible IDC is the most common form of breast cancer. About 80% of all breast cancers are IDC the cost of testing for breast cancer sets one back with $5000, which is a very huge amount for poor families. Therefore, detection of breast cancer using Histopathology images could reduce cost, time and accuracy of test. If you want to read more about breast cancer then read following article (<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5453426/>) as they used deep learning approach to study on histology images and achieved the sensitivity of 95 This shows the power of automation and how it could help in the detection of breast cancer.

***Literature Review:***

***Solution 1:***

In this Example CNN model is performed with 2-D Convolutional layer. In this example first layer has only 8 filters with 2 by 2. And the second layer have 4 by 4 size. Each layer has max pooling of size 2 by 2. And batch normalization is used with activation function relu this model have only 5 epochs for the classifications of IDC patches. The dataset has 20% of test data and 90% of training data with validation of 10%. The training and testing validation errors of 30,000 and 20,000 images.

***Link:***

[***https://www.kaggle.com/sarakaliman/simple-cnn-classification-model-with-acc-of-86***](https://www.kaggle.com/sarakaliman/simple-cnn-classification-model-with-acc-of-86)

***Result:***

This Model has accuracy of 86% however, model Sensitivity could be improved. Balanced accuracy is 80.8%.

***Solution 2:***

We will also make use of the IDC data set (a data set of the histology of breast cancer photos) by Kaggle. This data set contains a 2,77,524 50 x 50 patches, on the basis of 162 on Instagram, images of the breast cancer sample to be scanned at 40x magnification. 1 of them has a 98,738 negative tests, and 78,786 of the positive tests, the IDC. The dataset is available in the public domain. This requires, at the very least 3.02 GB of disk space.

***Link:***

[***https://data-flair.training/blogs/project-in-python-breast-cancer-classification/***](https://data-flair.training/blogs/project-in-python-breast-cancer-classification/)

***Result:***

In this example we learned to build a breast cancer classifier on the IDC dataset and created the network CancerNet for the same. We used Keras to implement the same.

***Solution 3:***

The Purpose of this core is to learn how to implement Has a good guide for all the tips and tricks to make the implementation of a CNN to get a better position, as a beginner. This is a popular approach in deep learning where pre-trained models are used as the basis for computer vision and natural language processing.

***Link:***

<https://www.kaggle.com/arbazkhan971/invasive-ductal-carcinoma-classification-89-acc>

***Result:***

In this project we get 77% Accuracy precision and recall. **CNN which is a specialized neural network for processing data that has an input shape like a 2D matrix like images.**

***Solution 4:***

**Building a Simple Machine Learning Model on Breast Cancer Data**

To begin this analysis, we asked whether we could improve on the results presented in the 2014 paper. For many years, and it's very likely that all of the methods that may be used in the work have been altered, improved, and further research has been done for you. However, it is a very good exercise for you to practice and develop your own deep learning, and data processing skills. Minimally invasive breast cancer (IDC) - in 95% of cases, and is one of the most common types of breast cancer.

***Link:***

[***https://towardsdatascience.com/building-a-simple-machine-learning-model-on-breast-cancer-data-eca4b3b99fa3***](https://towardsdatascience.com/building-a-simple-machine-learning-model-on-breast-cancer-data-eca4b3b99fa3)

***Result:***

So, in the end, we built a classification model, and we show that the random forest is a classification algorithm that provides the best possible results for our data. Well, this may not always be applicable to any data set. Select the model that you should always check with the analysis of the data, and then apply our machine learning model.

***Solution 5:***

# Analysis of Breast Cancer Detection Using Different Machine Learning Techniques

These algorithms show the positive effects of classification and encourage many researchers to use these types of algorithms to solve challenging tasks. In the convolutional neural network (CNN) it was used to predict and differentiate invasive ductal carcinoma from chest images with an accuracy of about 88%. In addition, data mining is widely used in the medical field to predict and differentiate rare phenomena to create a better understanding of any chronic diseases such as cancer.

***Link:***

[***https://link.springer.com/chapter/10.1007/978-981-15-7205-0\_10***](https://link.springer.com/chapter/10.1007/978-981-15-7205-0_10)

***Dataset Details:***

We will also make use of the IDC data set (a data set of the histology of breast cancer photos) by Kaggle. This data set contains a 2,77,524 50 x 50 patches, on the basis of 162 on Instagram, images of the breast cancer samples to be scanned at 40x magnification. 1 of them has a 98,738 negative tests, and 78,786 of the positive test, the IDC. The dataset is available in the public domain. This requires, at the very least 3.02 GB of disk space.

The names of the files in this dataset are as follows:

8863\_idx5\_x451\_y1451\_class0

Here 8863\_idx5, it is the patient's ID, 451, and in 1451, the x-and y-coordinates of the crop, and 0 is the class label of 0 indicates that there is not, IDC).

You will need to install a number of python packages are to be able to use these powerful python project. **The Dataset is a Supervised Learning and Multiclass Classification. Such as**

[**https://www.frontiersin.org/articles/10.3389/fgene.2019.00080/full**](https://www.frontiersin.org/articles/10.3389/fgene.2019.00080/full)

[**https://www.nature.com/articles/s41598-017-04075-z**](https://www.nature.com/articles/s41598-017-04075-z)

***Algorithm:***

***Pre - Processing:***

After importing the Dataset first, I count the pics then load the random images then load the images and set the target size of image 50/50 and get the pixels by dividing 255. Then by using numpy return the value of Stack. Then Load the images 90000 then plot the count plot by using seaborn. Then handling class imbalance problem then resize the target size again and set the pixels of images. Then load the images again and then after Pre-Processing check the Accuracy of load images by using Seaborn.

***Model Architecture and Details:***

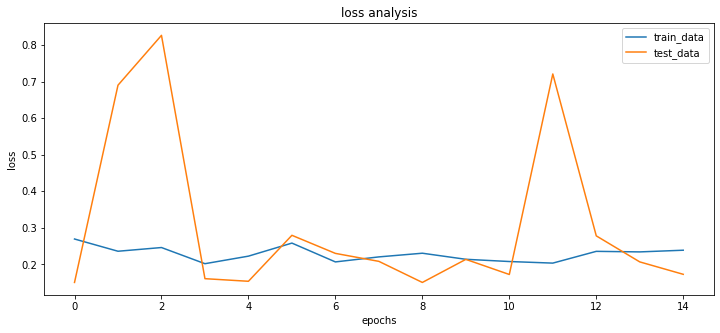
First, I install the Kaggle in colab then add application of Kaggle then make directory and add the dataset list then load the dataset and unzip all images in the colab. Then import the libraries and check the name of the directory inside which we have our files. Then Extract only png files now all the images will take up lot of memory, so therefore we will check until we get three different dimensions. These steps are necessary because the dataset is very big of 3 Gb and without these steps it is very difficult to load all data and get all images for training the model.

After loading the data in directory and after extract the images then plot the image by using images and cv2 directory then plot histogram of pixel values present in an image vs intensity. Then load the images in list classes and make column of class. Then calculate total images in class0 and class1 and display the percentage of images. Then read the images and train the model by using sklearn. model selection library. By using TensorFlow library categorize the trained model data and plot the data. Then get the train target and valid number of images by splitting the train test. Then get the shape of training testing and validation data. For getting the Weight and Loss I used following function

history=model.fit (x, epochs=15, validation\_split=0.3, batch\_size=56)

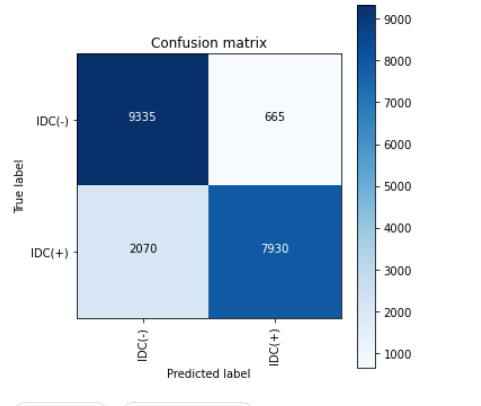
Then by using keras layers make the model by relu and softmax activation model. Then by check point save the best weights of the model for 20 epochs. Then make prediction and make confusion matrix and plot it. Then predict the X test and by using classification report get precision, recall, f1-score and Accuracy of the Model.

**Graph of loss Function:**



***Results and Conclusion:***

**Confusion Matrix:**

****

**Metrics:**  
[[9335 665] [2070 7930]]

Precision recall f1-score support

0 0.82 0.93 0.87 10000

1 0.92 0.79 0.85 10000

Accuracy 0.86 20000

macro avg 0.87 0.86 0.86 20000

weighted avg 0.87 0.86 0.86 20000