# **Breast Cancer Detection**

# 

**A PROJECT REPORT**

**Submitted by**

##### **Vedant Anand (19BAI10130)**

##### **Pratham Kurele (19BAI10163)**

##### **Manan Mittal (19BAI10171)**

##### **Devansh Rathi (19BAI10173)**

*In partial fulfilment for the award of the degree*

*Of*

##### **BACHELOR OF TECHNOLOGY**

*In*

**COMPUTER SCIENCE AND ENGINEERING**



**SCHOOL OF COMPUTING SCIENCE AND ENGINEERING**

**VIT BHOPAL UNIVERSITY**

**KOTHRIKALAN, SEHORE**

**MADHYA PRADESH - 466114**

**VIT BHOPAL UNIVERSITY, KOTHRIKALAN,**

**SEHORE**

**MADHYA PRADESH – 466114**

**BONAFIDE CERTIFICATE**

Certified that this project report titled “**Breast Cancer Detection” is** the Bonafide work of Manan Mittal (19BAI10171), Devansh Rathi (19BAI10173), Vedant Anand(19BAI10130), and Pratham Kurele (19BAI10163) who carried out the project work under my supervision. Certified further that to the best of my knowledge the work reported here does not form part of any other project / research work on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

**PROGRAM CHAIR PROJECT GUIDE**

**Dr. S. Southranjan** Dr. Sampath Kumar

School of Computer Science and School of Computer Science and . Engineering Engineering

VIT BHOPAL UNIVERSITY VIT BHOPAL UNIVERSITY

The Project Exhibition II Examination is held on \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**ACKNOWLEDGEMENT**

First and foremost, I would like to thank the Lord Almighty for His presence and immense blessings throughout the project work.

I wish to express my heartfelt gratitude to Dr. Manas Kumar Mishra, Head of the Department, School of Computing Science and Engineering for much of his valuable support encouragement in carrying out this work.

I would like to thank my internal guide Dr. Sampath Kumar, for continually guiding and actively participating in my project, giving valuable suggestions to complete the project work.

I would like to thank all the technical and teaching staff of the School of Computing Science and Engineering, who extended directly or indirectly all support.

Last, but not the least, I am deeply indebted to my parents who have been the greatest support while I worked day and night for the project to make it a success.

**LIST OF FIGURES (Table 1.1)**

|  |  |  |
| --- | --- | --- |
| **FIGURE NO.** | **TITLE** | **PAGE NO.** |
| **1** | **INTRODUCTION** | **10** |
| **2** | **Literature Survey** | **12** |
| **3** | **System Survey** | **13** |
| **4** | **Work Done** | **14** |
| **5** | **Disadvantages** | **15** |
| **6** | **Reference** | **15** |
| **7** | **Appendix-1** | **16** |
| **8** | **Appendix-2** | **21** |

**ABSTRACT**

Machine learning (ML) has become a vital part of medical imaging research. ML methods have evolved over the years from manual seeded inputs to automatic initializations. The advancements in the field of ML have led to more intelligent and self-reliant computer-aided diagnosis (CAD) systems, as the learning ability of ML methods has been constantly improving. More and more automated methods are emerging with deep feature learning and representations. Recent advancements of ML with deeper and extensive representation approaches, commonly known as deep learning (DL) approaches, have made a very significant impact on improving the diagnostics capabilities of the CAD systems.

**TABLE OF CONTENTS (Table1.2)**

|  |  |  |
| --- | --- | --- |
| **CHAPTER NO.** | **TITLE** | **PAGE NO.** |
|  | List of Figures  List of Tables  Abstract | iii  iv  v |
| 1 | **INTRODUCTION** 1.1 Introduction 1.2 Motivation for the work  1.3 About Introduction to the project  including techniques  1.4 Problem Statement  1.5 Objective of the work  1.6 Organization of the thesis  1.7 Summary | 10 |
| 2 | **LITERATURE SURVEY**  2.1 Introduction  2.2 <Core area of the project>  2.3 Existing Algorithms  2.4 Research issues from literature Survey  2.5 Summary | 12 |
| 3 | **SYSTEM ANALYSIS**  3.1 Introduction  3.2 Proposed System  3.3 Summary | 13 |
| 4 | **WORK DONE**  4.1 Introduction  4.2 Module 1  4.3 Module 2  4.4 Module 3  4.5 Module 4  4.6 Module 5  4.7 Module 6  4.8 Summary | 14 |
| 5 | **LIMITATIONS AND DISADVANTAGE** | 15 |
| **6** | **REFFERENCE** | 15 |
| 7 | **Appendix-1** | 16 |
| 8 | **Appendix-2** | 21 |

**INTRODUCTION**

**1.1** **Introduction :-**

Breast cancer is a dangerous disease for women. If it does not identify in the early-stage then the result will be the death of the patient. It is a common cancer in women worldwide. Worldwide near about 12% of women affected by breast cancer and the number is still increasing.

The doctors do not identify each and every breast cancer patient. That’s the reason Machine Learning Engineer / Data Scientist comes into the picture because they have knowledge of maths and computational power.

**1.2 Motivation for work :-**

There is only one way for our team to motivate our self to work hard: We don’t think about it as hard work. We think about it as part of making our self into who we want to be. Once we have made the choice to do something, We try not to think so much about how difficult or frustrating or impossible that might be; We just think about how good it must feel to be that, or how proud We might be to have done that. Make hard look easy.

**1.3** **Our project and techniques used :-**

This trains a neural network model to compare the Tissue and cells of a patient and a normal person . It's okay if you don't understand all the details; this is a fast-paced overview of a complete Tensor Flow program with the details explained.

**1.4** **Problem Statement :-**

To check whether a person is having a breast cancer or not.

**1.5** **Objective of the Work :-**

This review aimed to survey both traditional ML and DL literature with particular application for breast cancer diagnosis. The review also provided a brief insight into some well-known DL networks.

**1.7** **Summary :-**

From the report, it can be found that heterogeneous breast densities make masses more challenging to detect and classify compared with calcifications. The traditional ML methods present confined approaches limited to either particular density type or datasets. Although the DL methods show promising improvements in breast cancer diagnosis, there are still issues of data scarcity and computational cost, which have been overcome to a significant extent by applying data augmentation and improved computational power of DL algorithms.

**LITERATURE SURVEY**

**2.1 Introduction**

This research is done by the MIT because of the research in Machine Learning and Deep Learning.

We obtain this training model and because of this model we can predict the breast cancer in women.

**2.2 Existing Algorithms**

**Step** **1**: Start with Python compiler

**Step** **2**: Import Dataset

**Step 3**: Explore the data

**Step** **4**: Preprocess the data

**Step** **5**: Build the model

**Step** **6**: Train the model

**Step** **7**: Use the Trained model

**Step 8**: Stop

**2.3 Research issues/observations from**

There are many issues with the project because the Success ratios is only 90-95% sometimes it is above 90% because of the similar images.

* The Pixel of few images is very similar when it comes to the tissue so it is very difficult to distinguish between them.
* The success rate of this Almost 95% and sometimes it becomes very effective and give success rate 99%.

**SYSTEM ANALYSIS**

**3.1 Introduction :-**

Breast cancer (BC) is one of the most common cancers among women worldwide, representing the majority of new cancer cases and cancer-related deaths according to global statistics, making it a significant public health problem in today’s society.

The early diagnosis of BC can improve the prognosis and chance of survival significantly, as it can promote timely clinical treatment to patients. Further accurate classification of benign tumors can prevent patients undergoing unnecessary treatments. Thus, the correct diagnosis of BC and classification of patients into malignant or benign groups is the subject of much research. Because of its unique advantages in critical features detection from complex BC datasets, machine learning (ML) is widely recognized as the methodology of choice in BC pattern classification and forecast modelling.

**3.2 Proposed System**

First we will start the compiler and them we will import the dataset which helps us in making the project. Then we will import the kerels which is important to build and train the model. Then we will pre-process the data which we have to classify and then we have built the model in a way in which it can divide the images into pixels and then we compare each pixel to the infected tissue and then train the model to recognise the image and if again it encounter then it is easy to for computer to distinguish and give effective results

Then it will give the results that are shown by the program and hence our answer is there.

.

**WORK DONE**

**4.1 Module 1: - Importing breast cancer dataset from Kaggle**

The Dataset we have used here is the dataset of Wisconsin which help us predict whether the cancer is benign and malignant.

**4.2 Module 2: Explore the data**

We have divided the dataset into two parts that is benign and malignant.

Malignant is represented by 0 whereas benign is represented by 1.

After dividing we have checked the concentration of the tissue to determine whether it is contaminated or not . By this comparison we will predict the accuracy whether a group of women has breast cancer or not.

**4.3 Module 3: - Preprocess the data**

The data must be preprocessed before training the network. If you inspect the first image (tissue) in the training set, you will see that the pixel values fall in the range 0 to 255**.**

**4.4 Module 4: - Build the model**

**Applying the algorithm**

We have used ***Base line Algorithm*** and it consist of decision tree classifier, SVC, Gaussian NB, K-Neighbour Classifier. By using these algorithm in SVM we will predict the accuracy of the given dataset

**4.5 Module 5: - Use The Dataset**

With this algorithm we can save time of the doctors and the patient in the fast detection of breast cancer and solve it as soon as possible to minimise the risk of the death.

**DISADVANTAGES/LIMITATIONS IN THE SYSTEM**

* We cannot fully predict whether a women is having Brest cancer or not because of the different varieties of tissues and difference in genes.

**REFERENCES**

* machinelearningmastery.com
* tensorflow.google.cn
* en.wikipedia.org

**APPENDIX -1(Coding)**

**1) Import Modules:-**

mport numpy as np

import pandas as pd

import matplotlib.pyplot as plt

from sklearn.metrics import classification\_report

from sklearn.metrics import confusion\_matrix

from sklearn.metrics import accuracy\_score

from sklearn.model\_selection import train\_test\_split

from sklearn.model\_selection import cross\_val\_score

from sklearn.model\_selection import KFold

from sklearn.tree import DecisionTreeClassifier

from sklearn.neighbors import KNeighborsClassifier

from sklearn.naive\_bayes import GaussianNB

from sklearn.pipeline import Pipeline

from sklearn.preprocessing import StandardScaler

from sklearn.model\_selection import GridSearchCV

from sklearn.svm import SVC

import time

**2**) **Import the dataset:-**

data =pd.read\_csv(r"C:/Users/manan/dataset/pe.csv")

data.head(5)

**3) Explore the data:-**

data.describe()

data['diagnosis'] = data['diagnosis'].apply(lambda x: '1' if x == 'M' else '0')

data = data.set\_index('id')

del data['Unnamed: 32']

print(data.groupby('diagnosis').size())

data.plot(kind='density', subplots=True, layout=(5,7), sharex=False, legend=False, fontsize=1)

plt.show()

from matplotlib import cm as cm

fig = plt.figure()

ax1 = fig.add\_subplot(111)

cmap = cm.get\_cmap('jet', 30)

cax = ax1.imshow(data.corr(), interpolation="none", cmap=cmap)

ax1.grid(True)

plt.title('Breast Cancer Attributes Correlation')

# Add colorbar, make sure to specify tick locations to match desired ticklabels

fig.colorbar(cax, ticks=[.75,.8,.85,.90,.95,1])

plt.show()

Y = data['diagnosis'].values

X = data.drop('diagnosis', axis=1).values

X\_train, X\_test, Y\_train, Y\_test = train\_test\_split (X, Y, test\_size = 0.20, random\_state=21)

**4)** **Build the model:-**

models\_list = []

models\_list.append(('CART', DecisionTreeClassifier()))

models\_list.append(('SVM', SVC()))

models\_list.append(('NB', GaussianNB()))

models\_list.append(('KNN', KNeighborsClassifier()))

num\_folds = 10

results = []

names = []

for name, model in models\_list:

kfold = KFold(n\_splits=num\_folds, random\_state=123)

start = time.time()

cv\_results = cross\_val\_score(model, X\_train, Y\_train, cv=kfold, scoring='accuracy')

end = time.time()

results.append(cv\_results)

names.append(name)

print( "%s: %f (%f) (run time: %f)" % (name, cv\_results.mean(), cv\_results.std(), end-start))

**5)** **Train the model:-**

import warning

# Standardize the dataset

pipelines = []

pipelines.append(('ScaledCART', Pipeline([('Scaler', StandardScaler()),('CART', DecisionTreeClassifier())])))

pipelines.append(('ScaledSVM', Pipeline([('Scaler', StandardScaler()),('SVM', SVC( ))])))

pipelines.append(('ScaledNB', Pipeline([('Scaler', StandardScaler()),('NB', GaussianNB())]))

pipelines.append(('ScaledKNN', Pipeline([('Scaler', StandardScaler()),('KNN', KNeighborsClassifier())])))

results = []

names = []

with warnings.catch\_warnings():

warnings.simplefilter("ignore")

kfold = KFold(n\_splits=num\_folds, random\_state=123)

for name, model in pipelines:

start = time.time()

cv\_results = cross\_val\_score(model, X\_train, Y\_train, cv=kfold, scoring='accuracy')

end = time.time()

### results.append(cv\_results)

### names.append(name)

### print( "%s: %f (%f) (run time: %f)" % (name, cv\_results.mean(), cv\_results.std(), end-start))

fig = plt.figure()

fig.suptitle('Performance Comparison')

ax = fig.add\_subplot(111)

plt.boxplot(results)

ax.set\_xticklabels(names)

plt.show()

**6)** **Algorithm Tuning:-**

scaler = StandardScaler().fit(X\_train)

rescaledX = scaler.transform(X\_train)

c\_values = [0.1, 0.3, 0.5, 0.7, 0.9, 1.0, 1.3, 1.5, 1.7, 2.0]

kernel\_values = ['linear', 'poly', 'rbf', 'sigmoid']

param\_grid = dict(C=c\_values, kernel=kernel\_values)

model = SVC()

kfold = KFold(n\_splits=num\_folds, random\_state=21)

grid = GridSearchCV(estimator=model, param\_grid=param\_grid, scoring='accuracy', cv=kfold)

grid\_result = grid.fit(rescaledX, Y\_train)

print("Best: %f using %s" % (grid\_result.best\_score\_, grid\_result.best\_params\_))

means = grid\_result.cv\_results\_['mean\_test\_score']

stds = grid\_result.cv\_results\_['std\_test\_score']

params = grid\_result.cv\_results\_['params']

for mean, stdev, param in zip(means, stds, params):

print("%f (%f) with: %r" % (mean, stdev, param))

**7) prediction: -**

with warnings.catch\_warnings():

warnings.simplefilter("ignore")

scaler = StandardScaler().fit(X\_train)

X\_train\_scaled = scaler.transform(X\_train)

model = SVC(C=2.0, kernel='rbf')

start = time.time()

model.fit(X\_train\_scaled, Y\_train)

end = time.time()

print( "Run Time: %f" % (end-start))

with warnings.catch\_warnings():

warnings.simplefilter("ignore")

X\_test\_scaled = scaler.transform(X\_test)

predictions = model.predict(X\_test\_scaled)

print("Accuracy score %f" % accuracy\_score(Y\_test, predictions))

print(classification\_report(Y\_test, predictions))

**Appendix 2(coding)**

**1)** **Import libraries: -**

**2)** **Import the dataset:-**

****

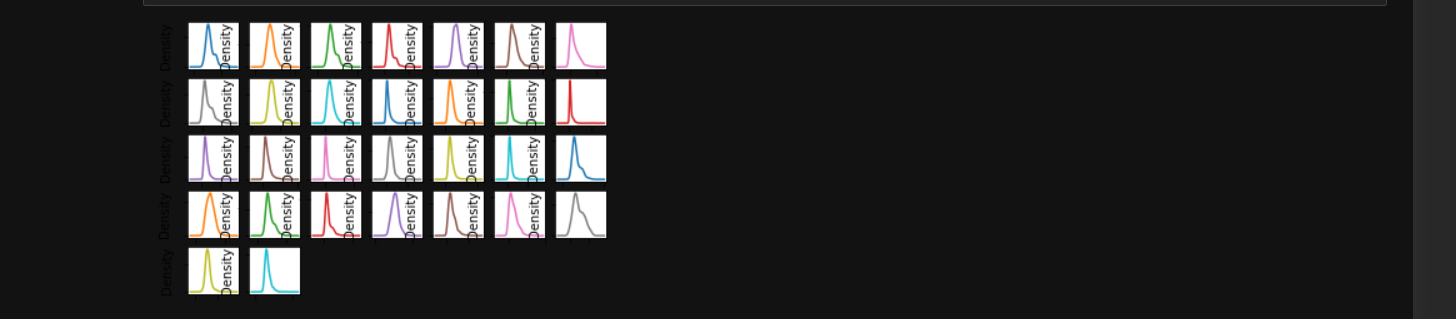
Fig. 1

1. **Divide the Dataset:-**

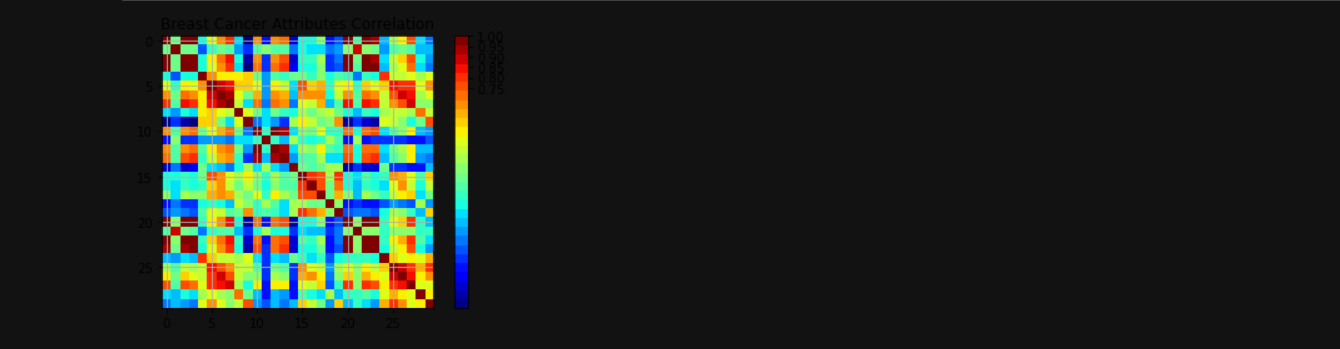


Fig. 2

**4) Preprocess the Dataset: -**

****

**Fig.3**

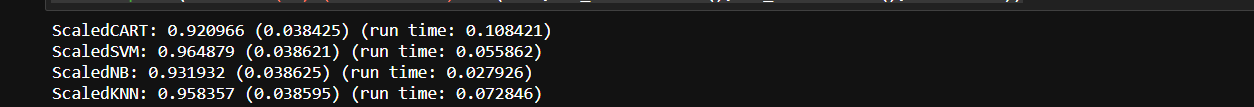


**Fig.4**

**5) Train the model:-**

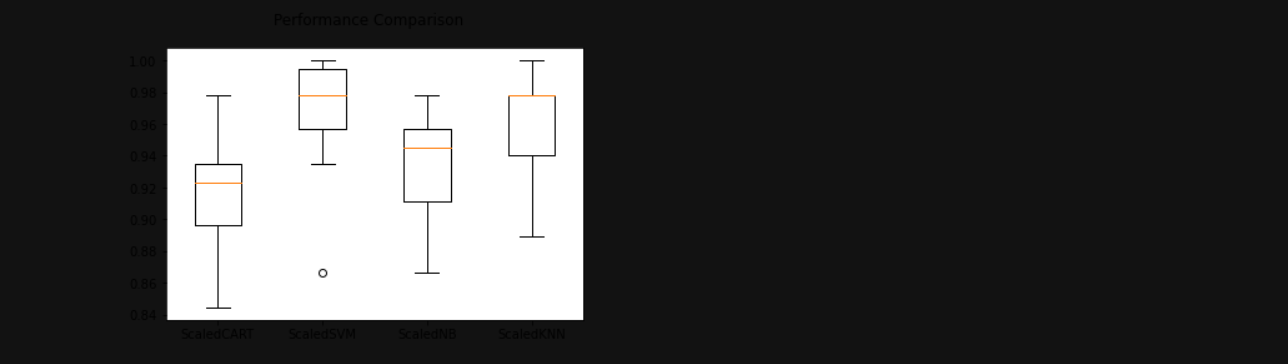


Fig.5

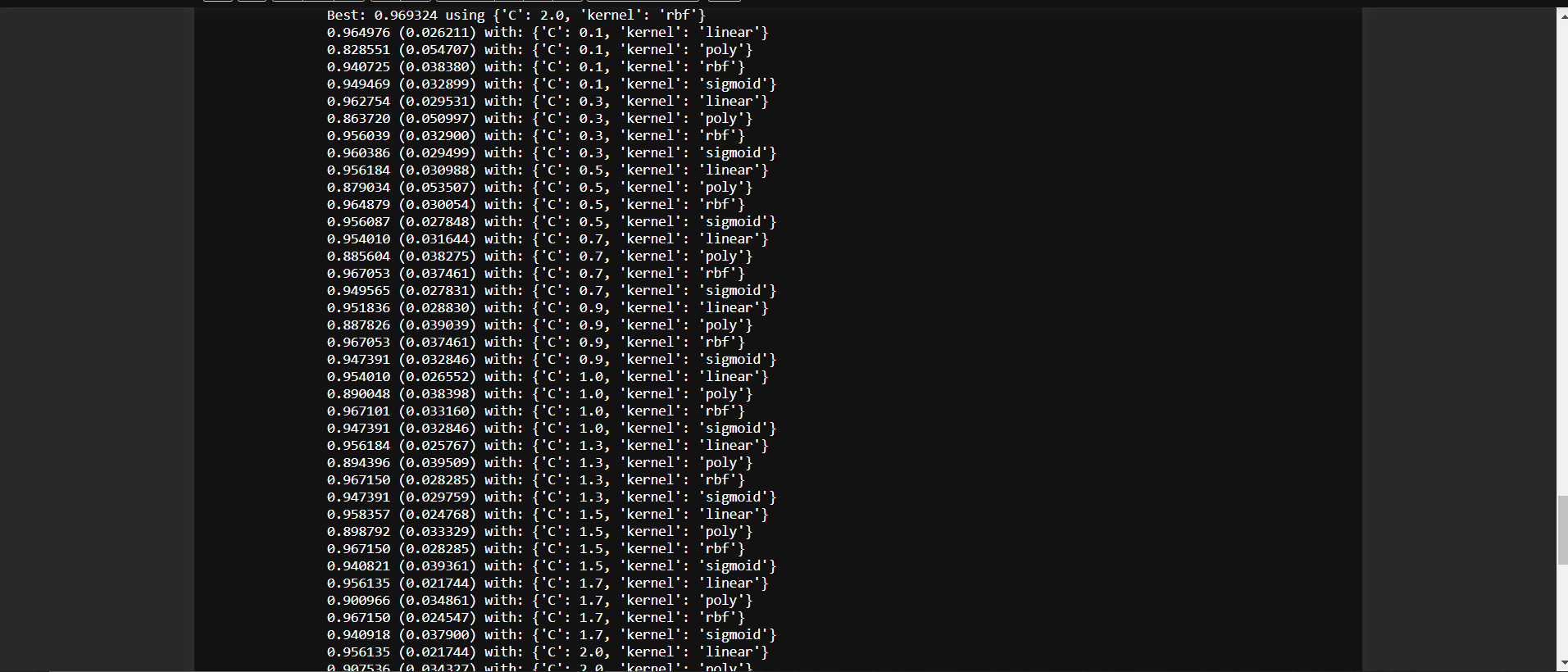


**Fig**.6

**6) Verify the predictions: -**

****

**Fig.7**

****

**Fig.8**

**7) Output: -**

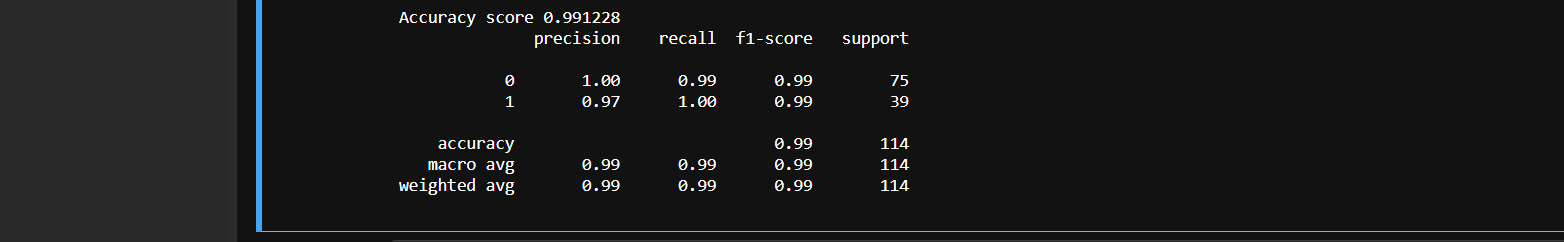
****

Fig.9