A MINI PROJECT REPORT
On
Breast Cancer Detection

Submitted by

Manmohan Agrahari
(161500309˘)

Ayush Gupta
(161500158)

Sujeet Kumar Singh
(161500569)

Mohd. Firoj
(161300052)

To
Mr. Amir Khan
Technical trainer, GLA University Mathura

Department of Computer Engineering & Applications
Institute of Engineering & Technology

GLA University
Mathura- 281406, INDIA

April, 2019

Table of Contents

Department of Computer Engineering and Applications
GLA University, Mathura
17 km. Stone NH#2, Mathura-Delhi Road, P.O. – Chaumuha,
Mathura – 281406

Declaration

*We hereby declare that the work which is being presented in the Mini Project "**Breast Cancer Detection",**in partial fulfillment of the requirements for Mini-Project LAB, is an authentic record of our own work carried under the supervision of **Mr. Amir Khan , Technical Trainer GLA University, Mathura**.*

Manmohan Agrahari

Sujeet Singh

Ayush Gupta

Mohd. Firoj

Department of Computer Engineering and Applications
GLA University, Mathura
17 km. Stone NH#2, Mathura-Delhi Road, P.O. – Chaumuha,
Mathura – 281406

CERTIFICATE

This is to certify that the project entitled ***"Breast Cancer Detection"*** *carried out in Mini Project – ILab is a bonafide work done by* ***Manmohan Agrahari (161500309), Sujeet Singh(161500569), Ayush Gupta (161500158) and Mohd.Firoj (161300052)****and is submitted in fulfillment of the requirements for the award of the degree Bachelor of Technology (Computer Science & Engineering).*

Signature of Supervisor:

Name of Supervisor:

Date:

ACKNOWLEDGEMENT

It gives us a great sense of pleasure to present the report of the B. Tech Mini Project undertaken during B. Tech. Third Year. This project in itself is an acknowledgement to the inspiration, drive and technical assistance contributed to it by many individuals. This project would never have seen the light of the day without the help and guidance that we have received.

Our heartiest thanks to ***Dr. (Prof). Anand Singh Jalal,*** *Head of Dept., Department of CEA for providing us with an encouraging platform to develop this project, which thus helped us in shaping our abilities towards a constructive goal.*

We owe special debt of gratitude to ***Mr.Amir Khan,*** *Technical Trainer Department of CEA, for his constant support and guidance throughout the course of our work. His sincerity, thoroughness and perseverance have been a constant source of inspiration for us. He has showered us with all his extensively experienced ideas and insightful comments at virtually all stages of the project & has also taught us about the latest industry-oriented technologies.*

We also do not like to miss the opportunity to acknowledge the contribution of all faculty members of the department for their kind guidance and cooperation during the development of our project. Last but not the least, we acknowledge our friends for their contribution in the completion of the project.

Manmohan Agrahari
Sujeet Singh
Ayush Gupta
Mohd. Firoj

Abstract

Breast cancer is the most common cause of death in women and the second leading cause of cancer deaths worldwide. Primary prevention in the early stages of the disease becomes complex as the causes remain almost unknown. However, some typical signatures of this disease, such as masses and micro classifications appearing on mammograms, can be used to improve early diagnostic techniques, which is critical for women's quality of life. X-ray mammography is the main test used for screening and early diagnosis, and its analysis and processing are the keys to improving breast cancer diagnosis. As masses and benign glandular tissue typically appear with low contrast and often very blurred, several computer-aided diagnosis schemes have been developed to support radiologists and internists in their diagnosis. In this article, an approach is proposed to effectively analyze digital mammograms based on texture segmentation for the detection of early stage tumors. The proposed algorithm was tested over several images taken from the digital database for screening mammography for cancer research and diagnosis, and it was found to be absolutely suitable to distinguish masses and micro classifications from the background tissue using machine learning techniques and a clustering algorithm for intensity-based segmentation.

1.Introduction

Breast cancer is the most common non-skin-related malignancy and the second leading cause of cancer death among women in the United States1 . Each year, more than 180,000 new cases of invasive breast cancer are diagnosed and more than 40,000 women die from the disease. Until research uncovers a way to prevent breast cancer or to cure all women regardless of when their tumor is found, early detection will be looked upon as the best hope for reducing the heavy toll of this disease. The early detection of cervical cancer by screening with the Papanicolaou smear (the Pap smear) dramatically reduced mortality from that cancer, and the rationale for the early detection of breast cancer is similar.

1.1Purpose

Machine learning is sub field of artificial intelligence is used to achieve thorough understanding of learning process and to implant learning capabilities in computer system it has various application in the areas of science, engineering and the society machine learning approach can provide generalized solution for a wide range of problem effectively and efficiently.the machine learning approaches makes computers more intelligent.Machine learning helps in solving prognostic and diagnostic problems in a variety of medical domain it is mainly used for prediction of disease progression,for therapy planning support and for overall patient management. Hypothesis from the patient data can be drawn from.We have witnessed that women are hesitant to visit a doctor due to fear of chemotherapy surgery and death negative publicity regarding pain during mammograph is prevalent .basically there is lack of motivation. while breast cancer can be diagnosed early using various app and tools.our project will help to detect breast cancer detection.

1.2 Scope

By using machine learning algorithm early breast cancer diagnosis will be helpful to know early cure and treatment,we will see plots for specific approaches showed how their performance depended on various parameters and the parameter values that showed higher performance are noted for overall comparison of approaches.Naive Bayes using kernel density estimation, K-NN classifier, Radial basis function networks and Support Vector machine showed high and almost equal accuracies.

Navies Bayes using normal distribution and multilayer perceptron showed lower accuracy when compared to others .As the training size increased the multilayer perceptron showed better accuracy than Naive Bayes using normal distribution than Naive Bayes using normal distribution. Naive Bayes using kernel density estimation,radial basis function networks and SVM showed slight decrease in theirs accuracies after reaching certain training data set size.The work in this thesis can be

extended by considering other approaches for comparison and finally the best one can be used to build a breast cancer diagnostic system with higher performance.

1.3 Overview

Developing an application using machine learning with python which aim or objective should promote a common goal for early detection down-staging Breast cancer detection and diagnoses to improve cancer outcomes.

This application will use machine learning concept using python .In machine Learning it will use KNN and SVM, KNN algorithm is one of the simplest classification algorithms. KNN is a non-parametric,lazy learning algorithm. Its purpose is to use a database in which the data points are separated into several

classes to predict the classification of a new sample point and SVM is supervised machine learning algorithm which can be used for classification or regression problems.it uses a technique called the kernel trick to transform your data

and then based on these transformations it finds an optimal boundary between the possible outputs.

2.Software Requirement Analysis

2.1 Why breast cancer detection?

The outlook for women with breast cancer has improved significantly since 1989 as the mortality rate has declines steadily a decline attribute both to earlier detection through wider use of a mammography screening and to improved treatments. Yet breast cancer remains a major problem second lung cancer cause of death from cancer for women this project aim is to provide and simple technique to predict breast cancer.

2.2 Modules and their functionalities:

2.2.1 Numpy:

NumPy is a module for Python. The name is an acronym for "Numeric Python" or "Numerical Python". It is an extension module for Python, mostly written in C. This makes sure that the precompiled mathematical and numerical functions and functionalities of Numpy guarantee great execution speed..Furthermore,NumPy enriches the programming language Python with powerful data structures, implementing multi-dimensional arrays and matrices.These data structures guarantee efficient calculations with matrices and arrays. The implementation is even aiming at huge matrices and arrays, better know under the heading of "big data". Besides that the module supplies a large library of high-level mathematical functions to operate on these matrices and arrays.

SciPy (Scientific Python) is often mentioned in the same breath with NumPy. SciPy needs Numpy, as it is based on the data structures of Numpy and furthermore its basic creation and manipulation functions. It extends the capabilities of NumPy with further useful functions for minimization, regression, Fourier-transformation and many others.Both NumPy and SciPy are not part of a basic Python installation. They have to be installed after the Python installation. NumPy has to be installed before installing SciPy.

Pandas:

Pandas is an open-source Python Library providing high-performance data manipulation and analysis tool using its powerful data structures. The name Pandas is derived from the word
Panel Data – an Econometrics from Multidimensional data.In 2008, developer WesMcKinney
started developing pandas when in need of high performance, flexible tool for analysis of data.Prior to Pandas, Python was majorly used for data munging and preparation. It had very little contribution towards data analysis. Pandas solved this problem. Using Pandas, we can accomplish five typical steps in the processing and analysis of data, regardless of the origin of data — load, prepare, manipulate, model, and analyze.Python with Pandas is used in a wide
range of fields including academic and commercial domains including finance, economics, Statistics, analytics, etc.

Key Features of Pandas:

- Fast and efficient DataFrame object with default and customized indexing.
- Tools for loading data into in-memory data objects from different file formats.
- Data alignment and integrated handling of missing data.
- Reshaping and pivoting of date sets.
- Label-based slicing, indexing and subsetting of large data sets.
- Columns from a data structure can be deleted or inserted.
- Group by data for aggregation and transformations.
- High performance merging and joining of data.
- Time Series functionality.

MATPLOTLIB:

Matplotlib is a python library used to create 2D graphs and plots by using python scripts. It has a module named pyplot which makes things easy for plotting by providing feature to control line styles, font properties, formatting axes etc. It supports a very wide variety of graphs and plots namely - histogram, bar charts, power spectra, error charts etc. It is used along with NumPy to provide an environment that is an effective open source alternative for MatLab. It can also be used with graphics toolkits like PyQt and wxPython.Conventionally, the package is imported into the Python script by adding the following statement MATPLOTLIB we can draw different-different graph such as.

- ❖ Line plots
- ❖ Data Distribution plots
- ❖ Discrete Data Plots
- ❖ Contour plots

Natural Language ToolKit (NLTK)

Natural Language processing is about developing applications and services that are able to understand human languages. a comprehensive Python library for natural language processing and text analytics. Originally designed for teaching, it has been adopted in the industry for research and development due to its usefulness and breadth of coverage. NLTK is often used for rapid prototyping of text processing programs and can even be used in production applications.

Key features of NLTK:-

1.Tokenize Text Using Pure Python

2.Count Word Frequency

3.Remove Stop Words Using NLTK

3.Software Design

3.1 Data Flow Diagram

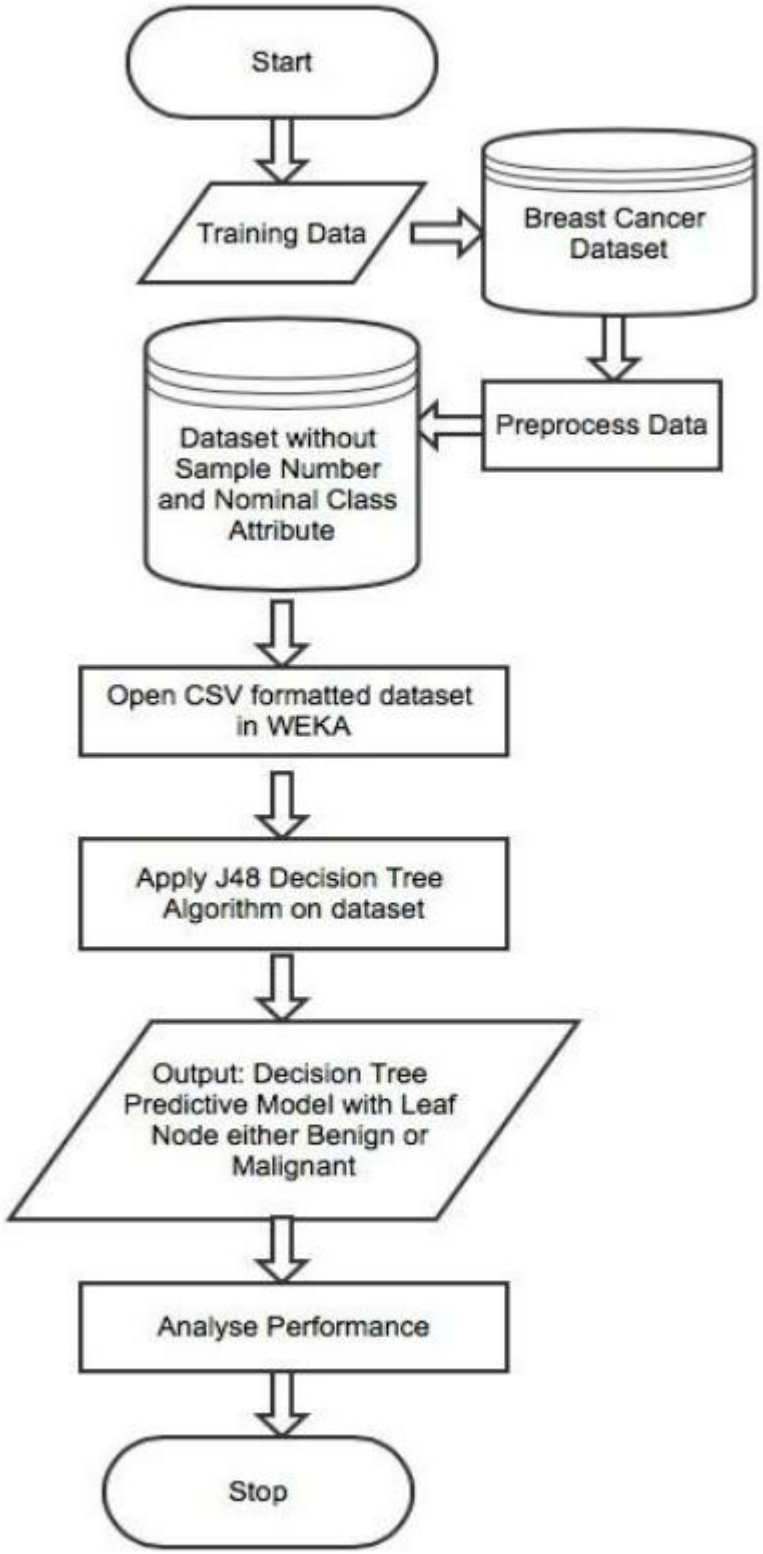

Fig. 3.1 Data flow diagram for Breast Cancer Detection

3.2 Sequence Diagram

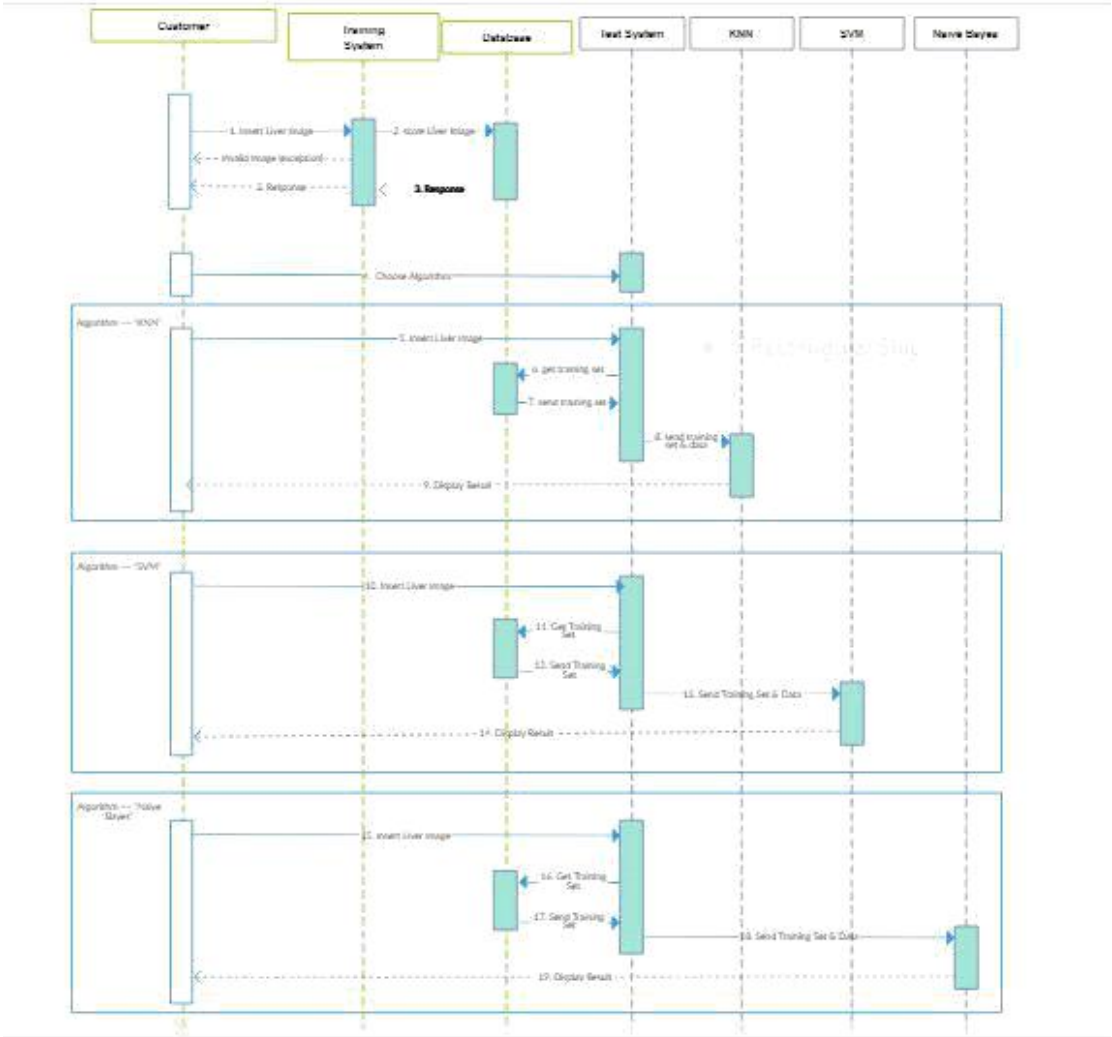

Fig. 3.2 Sequence diagram for Breast Cancer Detection

3.3 Use Case Diagram

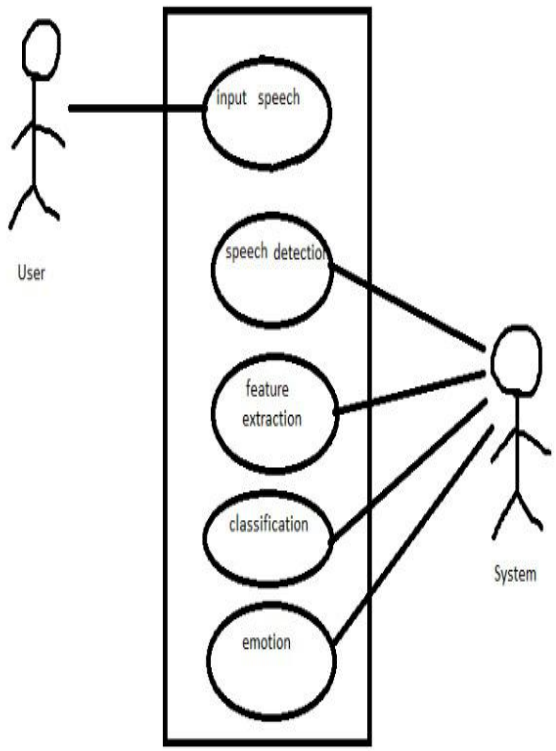

Fig. 3.3 Use case diagram for Breast Cancer Detection

4.Testing

4.1 Introduction:

Black box test cases

Parameters	Values	Output
clump thickness	5.0	
uniform cell size	1.0	
uniform cell shape	1.0	
marginal_adhesion	1.0	
single_epithelial_size	2.0	**2.0(Not Cancer)**
bare_nuclei	1.0	
bland_chromatin	1.0	
normal nuclei	1.0	
mitoses	2.0	

White box test cases

Parameters	Values	Output
clump thickness	8.0	
uniform cell size	10.0	
uniform cell shape	10.0	
marginal_adhesion	8.0	
single_epithelial_size	7.0	**4.0(Cancer)**
bare_nuclei	10.0	
bland_chromatin	9.0	
normal nuclei	7.0	
mitoses	1.0	

5.Implementation and User Interface

We have used Html , Css , Flask to make this system user friendly.

5.1 HTML

Hypertext Markup Language (HTML) is the standard markup language for creating web pages and web applications. With Cascading Style Sheets (CSS) and JavaScript, it forms a triad of cornerstone technologies for the World Wide Web.

Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML elements are delineated by *tags*, written using angle brackets. Tags such as <img /> and <input /> directly introduce content into the page. Other tags such as <p> surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags, but use them to interpret the content of the page.

HTML can embed programs written in a scripting language such as JavaScript, which affects the behavior and content of web pages. Inclusion of CSS defines the look and layout of content. The World Wide Web Consortium (W3C), maintainer of both the HTML and the CSS standards, has encouraged the use of CSS over explicit presentational HTML since 1997.

5.2 CSS

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language like HTML

CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.

CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts.

This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file, and reduce complexity and repetition in the structural content.

Separation of formatting and content also makes it feasible to present the same markup page in different styles for different rendering methods, such as on-screen, in print, by voice (via speech-based browser or screen reader), and on Braille-based tactile devices. CSS also has rules for alternate formatting if the content is accessed on a mobile device.

The name *cascading* comes from the specified priority scheme to determine which style rule applies if more than one rule matches a particular element. This cascading priority scheme is predictable.

The CSS specifications are maintained by the World Wide Web Consortium (W3C). Internet media type (MIME type) text/css is registered for use with CSS by RFC 2318 (March 1998). The W3C operates a free CSS validation service for CSS documents.

In addition to HTML, other markup languages support the use of CSS including XHTML, plain XML, SVG, and XUL.

5.3 Flask

Flask is a micro web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries.

It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. However, Flask supports extensions that can add application features as if they were implemented in Flask itself. Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies and several common framework related tools. Extensions are updated far more regularly than the core Flask program.

Flask is commonly used with MongoDB, which gives it more control over databases and history.

References/Bibliography

www.kaggle.com

www.udemy.com

www.w3school.com

6. Appendix

```
#Packages
import sys
import numpy
import matplotlib
import pandas
import sklearn

#version check
print('Python:{}'.format(sys.version))
print('Numpy:{}'.format(numpy.__version__))
print('matplotlib:{}'.format(matplotlib.__version__))
print('pandas:{}'.format(pandas.__version__))
print('sklearn:{}'.format(sklearn.__version__))


#importing packages
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from pandas.plotting import  scatter_matrix
from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsClassifier
from sklearn.model_selection import KFold
```

```
from sklearn.model_selection import cross_val_score

from sklearn.svm import SVC

from sklearn.metrics import classification_report,accuracy_score



#loading the dataset

names=['id','clump_thickness','univorm_cell_size','uniform_cell_shape','marginal_adh
esion','single_epithelial_size','bare_nuclei','bland_chromatin','normal_nuceloli','mitose
s','class']

dataset=pd.read_csv("data1.csv",names=names)



#taking care of missing data

dataset.replace('?',-99999,inplace=True)

print(dataset.axes)

dataset.drop(['id'],1,inplace=True)


#shape of dataset

print(dataset.shape)


# do dataset visualization

print(dataset.loc[6])

print(dataset.describe())


#plot histogram for each variable

dataset.hist(figsize=(10,10))

plt.show()
```

```
#create scatter plot matrix
scatter_matrix(dataset,figsize=(18,18))
plt.show()

#spliting into X and Y
X=dataset.iloc[:,:-1]
Y=dataset.iloc[:,9]

#splitting into training and test set
from sklearn.model_selection import train_test_split
X_train ,X_test,y_train,y_test=train_test_split(X,Y,test_size=0.2)

#specifying testing option
seed=8
scoring='accuracy'

#Define model to train
models=[]
models.append(('KNN',KNeighborsClassifier(n_neighbors=5)))
models.append(('SVM',SVC()))

#evaluate each model in turn
results=[]
names=[]
for name,model in models:
    kfold=KFold(n_splits=10,random_state=seed)
```

```
    cv_results=cross_val_score(model,X_train,y_train,cv=kfold,scoring=scoring)
    results.append(cv_results)
    names.append(name)
    msg="%s: %f (%f)" %(name,cv_results.mean(),cv_results.std())
    print(msg)


#make predictions on validation dataset
for name,model in models:
    model.fit(X_train,y_train)
    predictions=model.predict(X_test)
    print(name)
    print(accuracy_score(y_test,predictions))
    print(classification_report(y_test,predictions))

#now come to prediction at Example using KNN
clf=KNeighborsClassifier(n_neighbors=5)
clf.fit(X_train,y_train)
accuracy = clf.score(X_test, y_test)
print(accuracy)

#prediction on example
example= np.array([[4,2,1,1,1,2,3,2,5]])
example = example.reshape(len(example), -1)
prediction = clf.predict(example)
if(prediction==4):
    print("NOT CANCER")
```

```
if(prediction==2):
    print("CANCER")
```