Software Requirements Specification

for

BREAST CANCER DETECTION

Version 1.0 approved

Prepared by

MANMOHAN AGRAHARI(161500309)

SUJEET KUMAR SINGH(161500569)

AYUSH GUPTA(161500158)

MOHD. FIROJ(161300052)

GLA UNIVERSITY MATHURA

DATE-15/03/2019

Table of Contents

1. Introduction

1.1 Purpose

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 Document Convention

(I) Paper:A4

Font:

Times New Roman
12pt normal text,
16pt Bold for Chapter headings
14pt Bold for paragraph
headings 1.0 Line spacing

Header:

Upper left-Name of project Upper
right -Name of the chapter

Footer:

Lower Left	-Dept.of CEA,GLAU,Mathura
Lower Right	-Page No.

.

1.3 Product 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. The feature extraction process that is not considered in this thesis can be researched to extract better features for higher performance.

1.5 References

❖ Mousa, R.,Munib, Q., Moussam.,A., 2005. Breast cancer diagnosis system based on wavelet analysis and fuzzy-neural. Expert System Appl28, 713-723.

❖ Penna-Reyes, C.A., Sipper, M., 2000. A fuzzy genetic approach to breast cancer diagnosis. Artificial Intelligence Med. 17, 131-155.

❖ I. Kononenko, Machine learning for medical diagnosis: history, state of the art and perspective, Artificial Intelligence. Med. 23 (2001) 89109.

❖ Gardner, M. W., and Dorling, S. R. (1998). "Artificial neural networks (The multilayer perceptron) - A review of applications in the atmospheric sciences." Atmospheric Environment, 32(14/15), 2627-2636.

❖ Park J, Sand berg IW. Approximation and radial-basis-function networks. Neural Computation1993; 5: 305-16.

https://Google.com

❖ https://scikit-learn.org/stable/supervised_learning.html#supervised-learning

2.Overall description

2.1 Product Perspective

This product perspective is 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.

2.2 Product function

The main functionality of breast cancer detection to find out whether cancer is malignant or benign on the basis of the report.we will take some parameter and parameter value on the basis of that will try to find out whether it is malignant or benign. Malignant refers to cancer cells that can invade and kill nearby tissue and spread to other parts of your body.A benign is not malignant,which is cancer.It does not invade nearby tissue or spread to other parts of body the way cancer can.In most cases,the outlook with benign cancer is very good.But benign cancer can be serious if they press on vital structures such as blood vessels or nerves.Therefore,sometimes they require treatment and other times do not.

2.3 User Classes and Characteristic

In this project we tried to implement different-different algorithm such as Support Vector Machine K-NN. K-NN is stand for K-Nearest Neighbor. In pattern recognition, K Nearest Neighbor algorithm is a non-parametric algorithm that can be used for classification and regression. It defers the decision to generalize beyond the training examples till a new query is encountered. The training examples are represented as vectors in a multidimensional feature space and each example is labeled by a class. During the training phase the feature vectors and their class labels are stored. While in the classification phase where k is a user defined constant and the new unlabeled vector is assigned a class that is more frequent among its k nearest neighbors. The distance is calculated by one of the following measures.

2.4 Operating Environment

To operate this project the hardware and software component are required.

Software
Anaconda
Juptyer
Spyder

Hardware

i3-proceessor
4GB RAM

2.5 Design and Implementation Constraints

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.

2.6 User Documentation

(I) SVM

In machine learning, **support-vector machines** (**SVM**,also **support-vector networks**) are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis. Given a set of training examples, each marked as belonging to one or the other of two categories, an SVM training algorithm builds a model that assigns new examples to one category or the other, making it a non-probabilistic linear classifier (although methods such as Platt scaling exist to use SVM in a probabilistic classification setting). An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall.

In addition to performing linear classification SVM can efficiently perform a non-linear classification using what is called the kernel trick, implicitly mapping their inputs into high-dimensional feature spaces.When data is unlabelled,supervised learning is not possible, and an unsupervised learning approach is required, which attempts to find natural clustering of the data to groups, and then map new data to these formed groups. The **support-vector clustering** algorithm, created by Hava Siegelmann and Vladimir Vapnik, applies the statistics of support vectors, developed in the support vector machines algorithm, to categorize unlabeled data, and is one of the most widely used clustering algorithms in industrial applications.

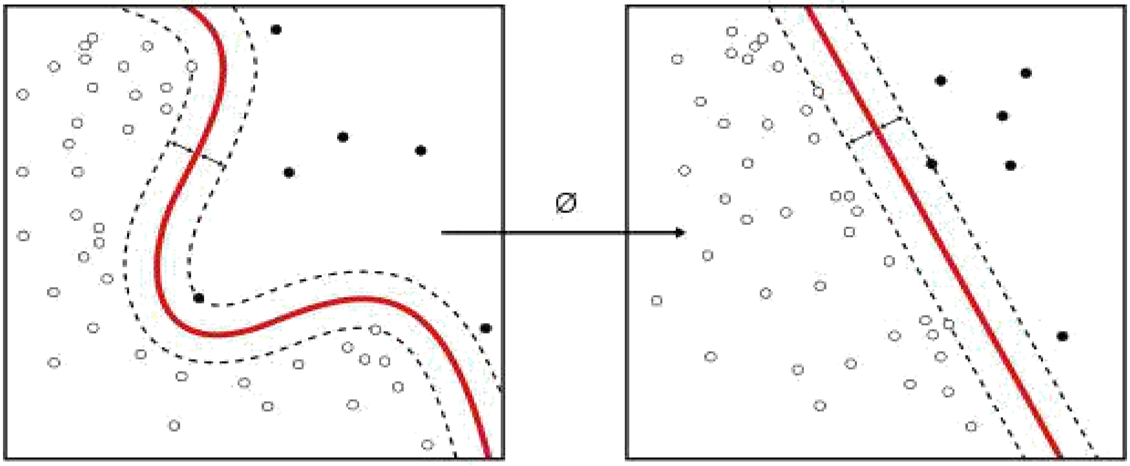

Fig.1 Machine Learning and data mining

(II) k-NN

In pattern recognition,the ***k*-nearest neighbors algorithm** (***k*-NN**) is a non-parametric method used for classification and regression.In both cases, the input consists of the *k* closest training examples in the feature space. The output depends on whether *k*-NN is used for classification or regression.In *k-NN classification*, the output is a class membership. An object is classified by a plurality vote of its neighbors, with the object being assigned to the class most common among its *k* nearest neighbors (*k* is a positive integer, typically small). If $k = 1$, then the object is simply assigned to the class of that single nearest neighbor.In *k-NN regression*, the output is the property value for the object. This value is the average of the values of its *k* nearest neighbors.

k-NN is a type of instance-based learning, or lazy learning, where the function is only approximated locally and all computation is deferred until classification. The *k*-NN algorithm is among the simplest of all machine learning algorithms.Both for classification and regression, a useful technique can be used to assign weight to the contributions of the neighbors, so that the nearer neighbors contribute more to the average than the more distant ones. For example, a common weighting scheme consists in giving each neighbor a weight of $1/d$, where *d* is the distance to the neighbor.The neighbors are taken from a set of objects for which the class (for *k*-NN classification) or the object property value (for *k*-NN regression) is known. This can be thought of as the training set for the algorithm, though no explicit training step is required.A peculiarity of the *k*-NN algorithm is that it is sensitive to the local structure of the data.

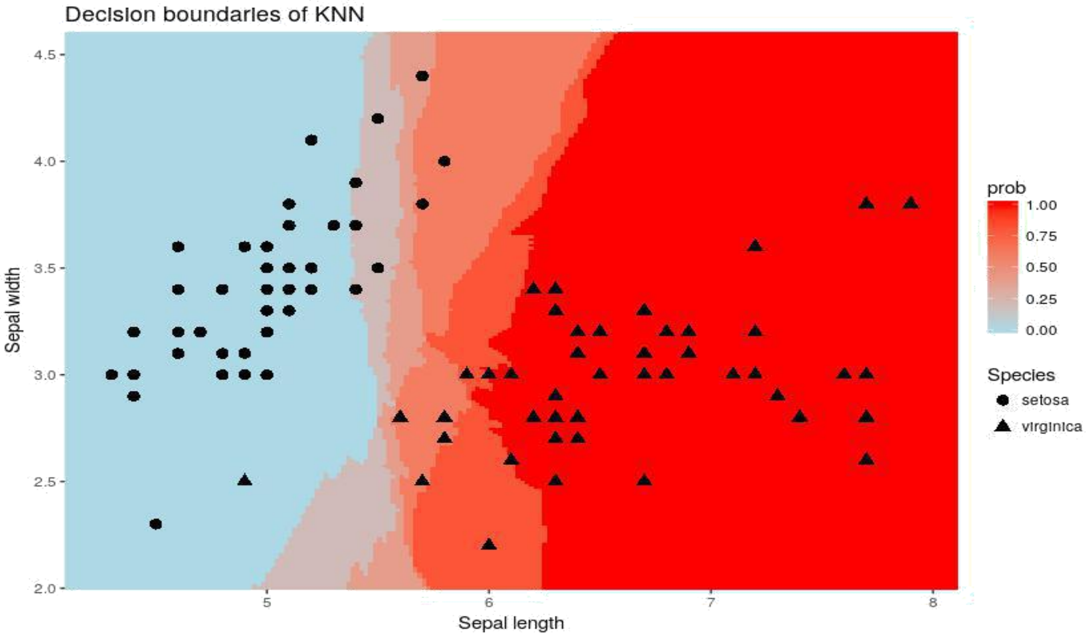

Fig.2 KNN Decision boundaries

(III). Cross validation:evaluating estimator performance

Learning the parameters of a prediction function and testing it on the same data is a methodological mistake: a model that would just repeat the labels of the samples that it has just seen would have a perfect score but would fail to predict anything useful on yet-unseen data. This situation is called **over fitting**. To avoid it, it is common practice when performing a (supervised) machine learning experiment to hold out part of the available data as a **test set** X_test, y_test. Note that the word “experiment” is not intended to denote academic use only, because even in commercial settings machine learning usually starts out experimentally. Here is a flowchart of typical cross validation work flow in model training. The best parameters can be determined by grid search techniques.In scikit- learn a random split into training and test sets can be quickly computed with the **train_test_split** helper function.

In statistics, over fitting is "the production of an analysis that corresponds too closely or exactly to a particular set of data, and may therefore fail to fit additional data or predict future observations reliably".An over fitted model is a statistical model that contains more parameters than can be justified by the data.The essence of over fitting is to have unknowingly extracted some of the residual variation (i.e. the noise) as if that variation represented underlying model structure.

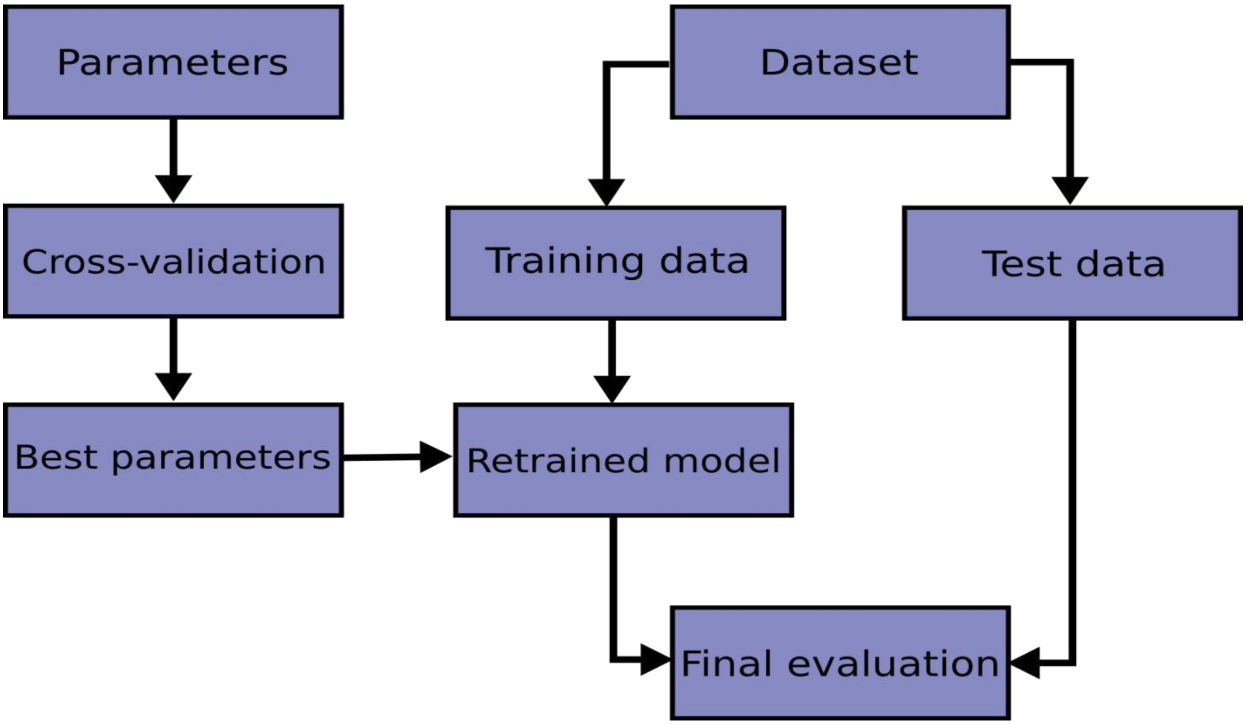

Fig 4. Cross validation:evaluating estimator performance

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:

Python is a great language for doing data analysis, primarily because of the fantastic ecosystem of data centric Python packages. Pandas is one of those packages, and makes importing and analyzing data much easier. Pandas builds on packages like NumPy and matplotlibto give you a single, convenient, place to do most of your data analysis and visualization work.

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

3. External Interface Requirements

3.1 User Interfaces:

Basically In user Interfaces we are providing GUI screening which gives the facility to its user to interact with application user basically have to give data set and corresponding result will be reflected on the next GUI page so by this way user can interact with our application and on GUI user also get instruction how he or she can interact with our application.Right now there would not be so much GUI page but in future number GUI screen can be extended

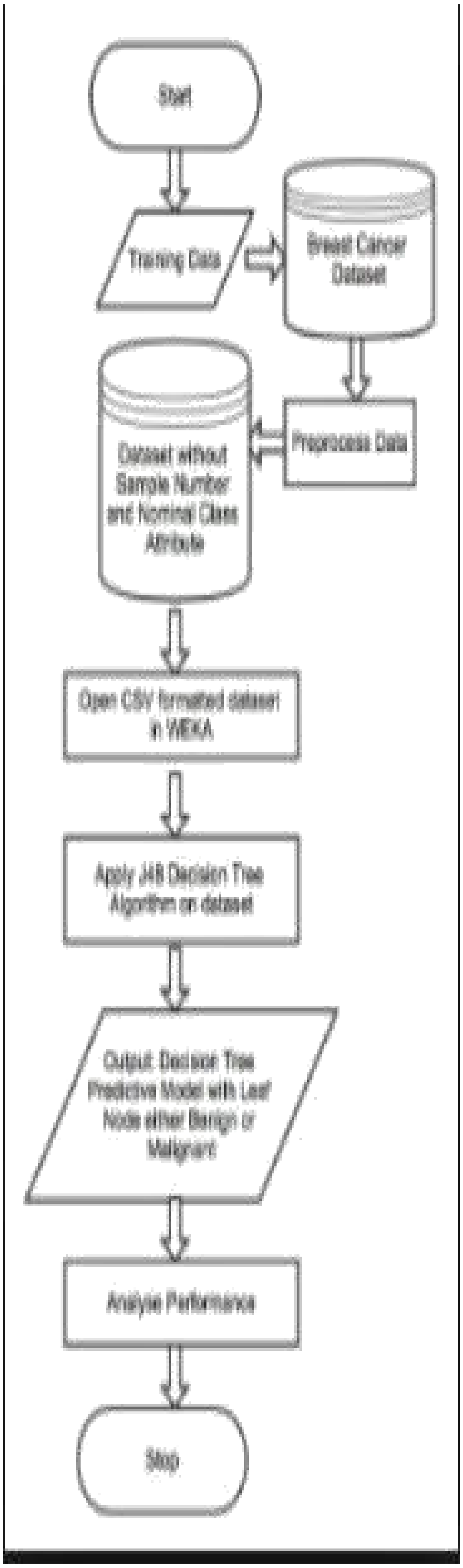
Start
Training Data
Breast Cancer Dataset
Preprocess Data
Dataset without Sample Number and Nominal Class Attribute
Open CSV formated dataset in WEKA
Apply J48 Decision Tree Algorithm on dataset
Output Decision Tree Predictive Model with Leaf Node either Benign or Malignant
Analyse Performance
Stop

Fig 3. flow chart

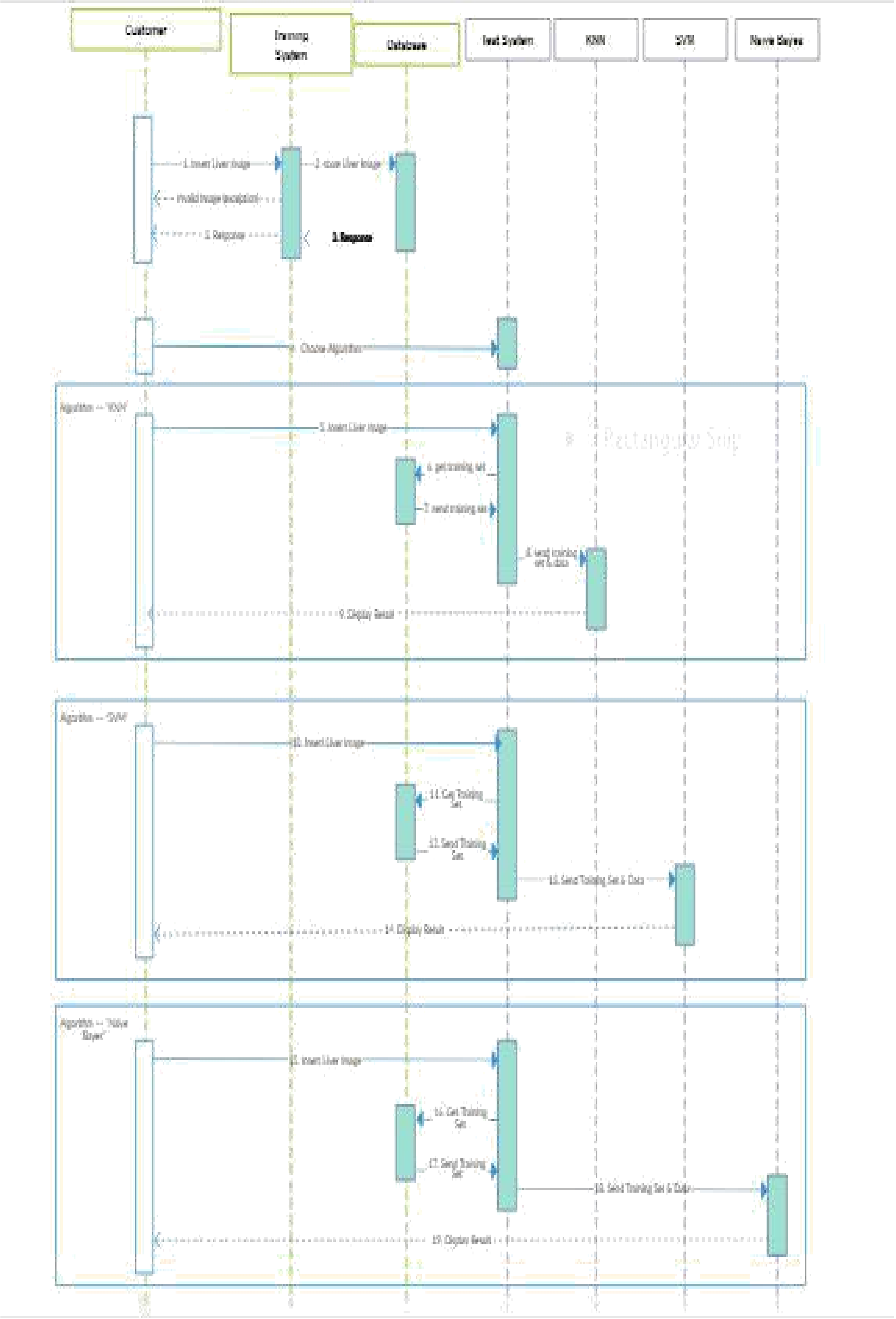

Fig 4. Sequence Diagram

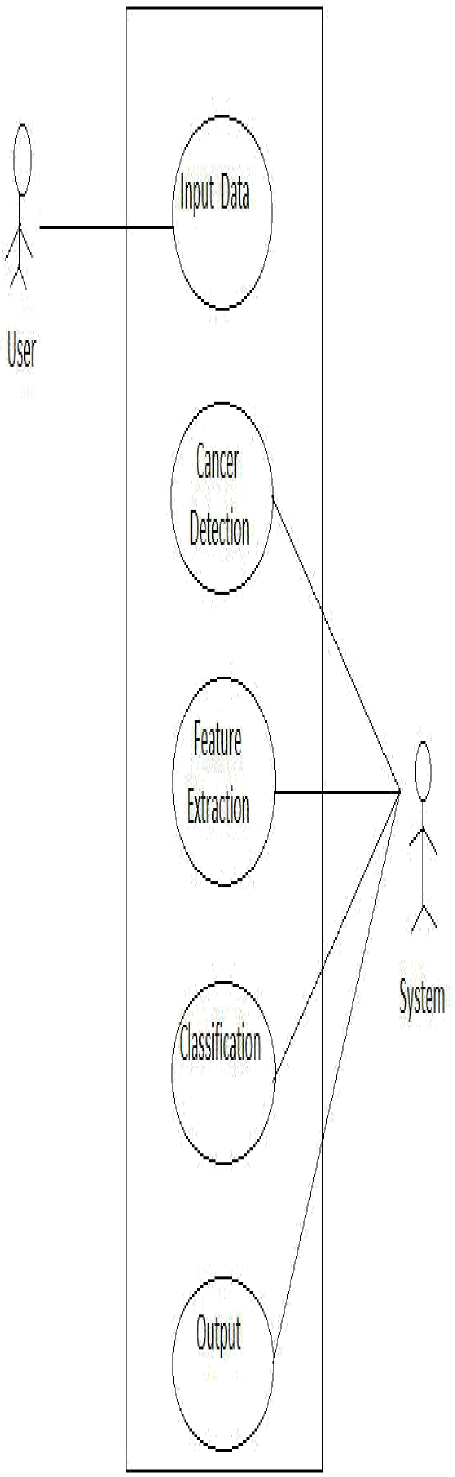

Fig 5. usecase diagram

3.2 Hardware Interfaces

In this project there is no need of hardware however some hardware are needed to run this project i3 processor having atleast 4GB Ram are needed to run this project smoothly. Right now these are requirement however in future when this project will extends there may need of more hardware to run this project smoothly.

3.3 Software Interfaces

The software which is needed to run this project are

- Anaconda
- Jupyter Notebook
- Spyder

Anaconda

Anaconda Navigator is a desktop graphical user interface (GUI) included in Anaconda distribution that allows users to launch applications and manage conda packages, environments and channels without using command-line commands. Navigator can search for packages on Anaconda Cloud or in a local Anaconda Repository, install them in an environment, run the packages and update them. It is available for Windows, macOS and Linux.

The following applications are available by default in Navigator :

JupyterLab
Jupyter Notebook
QtConsole
Spyder
Glueviz
Orange
Rstudio
Visual Studio Code

Jupyter Notebook

Jupyter Notebook (formerly IPython Notebooks) is a web-based interactive computational environment for creating Jupyter notebooks documents. The "notebook" term can colloquially make reference to many different entities, mainly the Jupyter web application, Jupyter Python web server, or Jupyter document format depending on context. A Jupyter Notebook document is a JSON document, following a versioned schema, and containing an ordered list of input/output cells which can contain code, text (using Markdown), mathematics, plots and rich media, usually ending with the ".ipynb" extension.

A Jupyter Notebook can be converted to a number of open standard output formats (HTML, presentation slides, LaTeX, PDF, ReStructuredText, Markdown, Python) through "Download As" in the web interface, via the nbconvert library or "jupyter nbconvert" command line interface in a shell.To simplify visualisation of Jupyter notebook documents on the web, the nbconvert library is provided as a service through NbViewer which can take a URL to any publicly available notebook document, convert it to HTML on the fly and display it to the user.Jupyter Notebook provides a browser-based REPL built upon a number of popular open-source libraries:

IPython
ØMQ
Tornado (web server)
jQuery
Bootstrap (front-end framework)
MathJax

Spyder

Spyder is an open source cross-platform integrated development environment (IDE) for scientific programming in the Python language. Spyder integrates with a number of prominent packages in the scientific Python stack,including NumPy, SciPy, Matplotlib, pandas, IPython, SymPy and Cython, as well as other open source software.It is released under the MIT license.Initially created and developed by Pierre Raybaut in 2009, since 2012 Spyder has been maintained and continuously improved by a team of scientific Python developers and the community.

Spyder is extensible with first- and third-party plugins,includes support for interactive tools for data inspection and embeds Python-specific code quality assurance and introspection instruments, such as Pyflakes, Pylint and Rope. It is available cross-platform through Anaconda, on Windows, on macOS through MacPorts, and on major Linux distributions such as Arch Linux, Debian, Fedora, Gentoo Linux, openSUSE and Ubuntu.Spyder uses Qt for its GUI, and is designed to use either of the PyQt or PySide Python bindings.QtPy, a thin abstraction layer developed by the Spyder project and later adopted by multiple other packages, provides the flexibility to use either backend.

3.4 Communications Interfaces

In this project there is no requirement of FTP and HTTP protocol because this project is not running on real time data we are doing prediction on static data.that’s why do not need any kind FTP and HTTP protocols.

4. System Features

The features of the system is to detect cancer by just providing some of the parameters like marginal adhesion,clump thickness,uniform cell size etc.There are two results for this project one is Malignant which means having cancer and the other is Benign which means not cancer it depends on the parameters which may lead to a result of Malignant or Benign.

4.1 Usefulness of SVM and KNN

4.1.1 Description and Priority

The best feature of the project is that it uses two machine learning algorithms first is SVM which is support vector machine and the other is KNN which is K-nearest neighbors .Talking about the priority these two of the algorithms are very important for the project.One can't be neglected at the cost of other. So talking about priority wise both are important and must be at a scale of 9.

4.1.2 Stimulus/Response Sequences

The activities involves various things.The first thing is that this project is a user interface project so it requires taking input from user by providing various parameters by the user then it needs to fetch that parameters by taking them into a file and then match with the data set and then provide result of benign and malignant.

4.1.3 Functional Requirements

The system requires some parameters which are eight in no. if the parameters are specified more then the required parameters then surely it will result into an error or there can be a case that the no. of parameters provided can be less then the required parameters then in that case too it will result into an error.The main requirement is that to provide only eight no. of parameters .

5. Other Nonfunctional Requirements

5.1 Performance Requirements

The product is accurate with a accuracy percentage of approx 95 .This accuracy can further be increased if there is more clear data for this project..If there is a data set for with each content correctly and accurately specified then the accuracy can be increased . A Support Vector Machine (SVM) is a discriminative classifier formally defined by a separating hyper plane. In other words, given labeled training data (supervised learning), the algorithm outputs an optimal hyper plane which categorizes new examples. K nearest neighbors is a simple algorithm that stores all available cases and classifies new cases based on a similarity measure (e.g., distance functions). k-NN has been used in statistical estimation and pattern recognition.

Safety Requirements:

This project does not lead to any harm even it can detect cancer if a person is having.The best part of this project is that if a person is feared about it. So just providing his parameters he will we able to know about his cancer at the right time he can have its diagnosis as soon as possible.

5.2 Security Requirements

There are no security issues with the project as this project is secured just need to provide the data of any patient and it will detect whether the patient is having cancer or not .If cancer is there it will result malignant which means cancer otherwise it will result Benign which means not cancer.One more important thing is the data set which should be precise and accurate .

5.3 Software Quality Attributes

Cancer is the second-most common cause of death (after cardio-vascular disease) . So its very important to detect it because right diagnosis at right time is very important and that’s what this project does . this project quality is to help in detecting cancer which is a major concern of this project.

5.4 Business Rules

The Business rules for this project are actually quite simple .Just one need to give some of the parameters which can be known by performing certain activities like clump thickness,uniform cell size uniform cell shape,marginal adhesion etc just after providing these parameters it will detect cancer is malignant or benign .

6. Other Requirements

The project does not require any use of database.The project objective is just to detect cancer . The main thing of project is that it needs a data set which must be precise and clear then only it will be getting the best accuracy of the project .So most important factor is data set. This project will be using the machine learning algorithms first is Support Vector Machine (SVM) and KNN(k-nearest neighbor).

Appendix A: Glossary

SVM : Support vector Machine

k-NN : k-nearest neighbor