جامعة الشرق الأوسط
MIDDLE EAST UNIVERSITY
Amman - Jordan

# Tuning Approach to Improve Multilayer Perceptron for Breast Cancer Prediction

# أسلوب ضبط لتحسين التنبؤ بسرطان الثدي باستخدام قاعدة الادراك متعددة الطبقات

**Prepared by**
**Fida'a Mousa Ahmad Al-Shami**

**Supervisor**
**Dr. Bassam Al-Shargabi**

**A Thesis Submitted in Partial Fulfillment of the Requirements for the Master Degree in Computer Science**

**Department of Computer Science**

**Faculty of Information Technology**

**Middle East University**

**June, 2019**

# Authorization

I, **Fida'a Mousa Ahmad Al-Shami**, authorize Middle East University to provide Libraries, Organizations, and Individuals with Copies of my Thesis on Request.

Name: Fida'a Mousa Ahmad Al-Shami.

Date: 17/6/2019

Signature:

# Examination Committee Decision

This is to certify that the thesis entitled "Tuning Approach to Improve Multi-Layer Perceptron Breast Cancer Prediction" was successfully defended and approved on 1/6/2019.

| Examination Committee Members | Signature |
| --- | --- |
| (*Supervisor*)<br>**Dr. Bassam Al-Shargabi**<br>*Associate Professor, Department of Computer Science*<br>*Middle East University* | ٢٠١٩/٦/١٨ |
| *(Chairman of Examination Committee and Internal Committee Member)*<br>**Dr. Hesham Abusaimeh**<br>*Associate Professor, Department of Computer Science*<br>*Middle East University* | 2019/6/17 |
| *(External Committee Member)*<br>**Dr. Thamer Al-Rousan**<br>*Associate Professor, Department of Computer Science*<br>*Isra University* | ٢٠١٩/٦/١٧ |

## Acknowledgement

This research was developed by the grace of Allah who gave the researchers the knowledge and wit to finish and established this research entitled **Tuning Approach to Improve Multilayer Perceptron for Breast Cancer Prediction**

The researcher would like to thank the following people who helped to make this study possible. To **DR. Bassam Al-Shargabi** who patiently taught me everything, she needs to know. Thank you I appreciate from the bottom of my heart. The researcher appreciates everything you have done .

And now the researcher would like to thank all the people who supported her, she's family, friends, and classmates. Especially her Mum who helped her with everything.

Finally, to her friends who believed that they can finish the study despite all the struggles, depression, and stress they experienced in the making of this research.

She doesn’t forget who sparked the challenge of science and knowledge. **Dr.Ahmad Abu-Shareha** for what he has given her from science and she wish him good luck wherever he goes. She also thanks all of her doctors from the Department of Information Technology, the dean **Dr Abdelrhman AbuArqoub**, **Prof. Hamza Al-Sewadi**, **Dr Hisham Abu Saaymeh** . Special thanks to the doctor who is full of science, humility and sophistication, **Dr Mudhafar Al-Jarrah** on every knowledge which was presented, **Dr Rami Alkhawaldeh** who spent his time, effort, and knowledge as soon as needed.

From the researcher a VERY BIG THANKS to YOU to all of you and ALLAH BLESS!!!!

**The Researcher**

# Dedication

This study is whole-heartedly dedicated to my beloved parents, who have been my source of inspiration and gave me strength when I thought of giving up, who continually provide their moral, spiritual, emotional, and financial support .

To my brothers, sisters, relatives, mentor, friends, and classmates who shared their words of advice and encouragement to finish this study .

Lastly, I dedicated this thesis to the Almighty Allah, thank you for the guidance , strength, power of the mind, protection and skills and for giving us a healthy life. All of these, I offer to you.

**The Researcher**

# Table of Contents

# List of Figures

# List of Tables

# List of Abbreviations

| Abbreviations | Meaning |
|---|---|
| ACC | Accuracy |
| ACS | American Cancer Society |
| AI | Artificial Intelligence |
| ANN | Artificial Neural Networks |
| AUC | Area Under Curve |
| B | Benign |
| BC | Breast Cancer |
| DL | Deep Learning |
| DTs | Decision Trees |
| FM | F-Measure |
| FS | Feature Selection |
| IHC | Immuno Histo Chemical |
| KNN | K Nearest Neighbor |
| M | Malignant |
| ML | Machine Learning |
| MLP | Multi-Layer Perceptron |
| P | Precision |
| R | Recall |
| RF | Random Forest |
| ROC | Receiver Operating Characteristic |
| SVM | Support Vector Machines |
| WDBC | Wisconsin Dataset Breast Cancer |

# Tuning Approach to Improve Multi-Layer Perceptron Breast Cancer Prediction

**Prepared By: Fida'a Al-Shami**

**Supervisor: Dr. Bassam Al-Shargabi**

## Abstract

Breast Cancer has turned into a typical disease around the globe in young women and the main source of cancer death and caused 22.9% of a wide range of cancers in women. The development of massive breast cancer screening has led to earlier diagnosis and rapid management with a significant improvement in survival rate. The problem of automatically searching for information contained in medical images is extremely needed. There is difficulty in interpretation these images as well as their large numbergenerates tedious work for those who must interpret them. In order to process this large volume of information, doctors are currently turning to the use of systems to assist in the analysis and interpretation of these images. This could be achieved by Machine Learning (ML) techniques.

The proposed approach in this thesis compares the ML algorithms (KNN, RF and MLP) of the classification on the breast cancer WDBC dataset to obtain the best algorithms in the results. Then it identifies the most specific and relevant attributes of the malignant tumour classification through more than one feature selection algorithms. Next, it determines which activation function for MLP that produces a more accurate result. Finally, tuned MLP and comparing evaluation metrics in machine learning for original MLP and tuned MLP.

The experimental results over WDBC dataset showed that the accuracy enhancement of proposed Tuned MLP compared to the original MLP is around 1.07%.

**Keywords: Machine-learning classification, Evaluation Metrics, Feature selection, Breast cancer, WDBC.**

## أسلوب ضبط لتحسين التنبؤ بسرطان الثدي باستخدام قاعدة الادراك متعددة الطبقات

**إعداد**

**فداء موسى الشامي**

**إشراف**

**الدكتور بسام الشرجبي**

## الملخص

تحول سرطان الثدي إلى مرض شائع في جميع أنحاء العالم عند النساء الشابات والمصدر الرئيسي لوفيات السرطان وتسبب في 22.9 ٪ من مجموعة واسعة من أنواع السرطان لدى النساء. أدى تطور الفحص الشامل لسرطان الثدي إلى التشخيص المبكر والإدارة السريعة مع تحسن كبير في معدل البقاء على قيد الحياة. هناك حاجة ملحة لمشكلة البحث التلقائي عن المعلومات الموجودة في الصور الطبية. فهناك صعوبة في تفسير هذه الصور بالإضافة إلى أعدادها الكبيرة، مما يولد عملاً شاقًا لأولئك الذين يجب عليهم تفسيرها. لمعالجة هذا الحجم الكبير من المعلومات، يلجأ الأطباء حاليًا إلى استخدام الأنظمة للمساعدة في تحليل وتفسير هذه الصور. يمكن تحقيق ذلك من خلال تقنيات التعلم الآلي.

يقارن النهج المقترح في هذه الأطروحة خوارزميات التعلم الآلي (خوارزمية أقرب جار، خوارزمية الغابة العشوائية، وخوارزمية قاعدة الادراك متعددة الطبقات) للتصنيف على قاعدة بيانات سرطان الثدي WDBC للحصول على أفضل الخوارزميات في النتائج. ثم تحديد السمات الأكثر تحديدًا والأكثر صلة بتصنيف الورم الخبيث من خلال أكثر من خوارزميات اختيار الميزة.

بعد ذلك، تقييم النتائج حسب المقاييس الخاصة بالتقييم على نتيجة قاعدة الادراك متعددة الطبقات للوصول إلى نتيجة أكثر دقة. وأخيرًا، مقارنة نتائج مقاييس التقييم في التعلم الآلي على قاعدة الادراك متعددة الطبقات المعدل ونتائج قاعدة الادراك متعددة الطبقات الأصلي.

أظهرت النتائج التجريبية على مجموعة بيانات WDBC أن تحسين دقة قاعدة الادراك متعددة الطبقات المعدل المقترحة مقارنة بـ قاعدة الادراك متعددة الطبقات الأصلي يبلغ حوالي 1.07٪.

**الكلمات المفتاحية: تصنيف تعلم الآلة، مقاييس التقييم، اختيار الميزات، سرطان الثدي.**

# Chapter One

# Introduction

# Chapter One
# Introduction

## 1.1 Introduction

According to the American Cancer Society (ACS), worldwide studies reported 627,000 deaths from breast cancer in 2018 and this number is expected to increase impacting 2.1 million women in 2019. Breast cancer has turned into a typical disease around the globe caused 22.9% of a wide range of cancers in women and the main source of cancer death (Duijm et al., 2004). Furthermore, about 10% of women have breast cancer in western nations, millions of women experience the ill effects of this weakening dangerous disease (Duijm et al., 2004).

Machine learning techniques develop prediction models for classifying future occasions in a manner steady with historical information. These techniques have often shown up in different practice regions and particularly healthcare and biology. Classification and prediction issues have an essential role in medical decision making (Pendharkar et al., 1999) and, in this manner, because of illnesses diagnosis significance to humankind, a few examinations have been led on modelling procedures for their classification (Karabatak, 2015; Zare-Zardini et al., 2015)

The development of massive breast cancer screening has led to earlier diagnosis and rapid management with a significant improvement in survival rate. The treatment and analysis of medical images is a rapidly expanding area. The problem of automatically searching for information contained in medical images is urgently needed. Indeed, the great diversity of medical imaging devices, the difficulty of interpretation of these images as well as their large number, generates tedious work for those who must interpret them. (Zheng, Yu, & Kambhamettu, 2007)

In order to process this large volume of information, doctors are currently turning to the use of systems to assist in the analysis and interpretation of these images. This analysis aims to facilitate the diagnosis made by the practitioner and to make it as accurate and reliable. However, and in contrast to advanced technology in the medical sector, breast cancer analysis remains a real public health problem and a very sensitive. This could be achieved by Machine Learning (ML) techniques.

Machine Learning (ML) is one of Artificial Intelligence (AI) fields that imitates the workings of the human brain in processing data and creating patterns for use in decision making. Machine learning has networks capable of learning can be supervised, semi-supervised or unsupervised from data.

In this study, which is based on ML, we exploit Multi-Layer Perceptron (MLP) to improve prediction to the prediction of cancer tumours in the early stages and reduce the mortality of cancer, which is the second leading cause of death in the world in 2015, killing 8.8 million people. (Organisation, 2004)

## 1.2 Background of the study

It is important at the beginning to clarify the exact meaning of the common terms used in the field of the study are briefly defined as follows:

**- Machine learning:**

The term machine learning intends to enable machines to learn without programming them expressly. There are four general machine learning methods: (1) supervised, (2) unsupervised, (3) semi-supervised, and (4) reinforcement learning methods.

The goals of machine learning are to enable machines to make predictions, perform clustering, extract association rules, or make decisions from a given dataset. (Mohammed, Khan, & Bashie, 2016)

**- Breast Cancer:**

It begins when cells in the breast is increase out of control. These cells generally structure a tumour that can frequently be seen on an x-ray or felt like a bump. The tumour is malignant (cancer) if the cells can develop into (attack) encompassing tissues or spread to distant areas of the body. (Facts, 2017)

**- Feature selection:**

Feature selection is one of the central ideas in machine learning which hugely impacts the performance of your model.

Also, it is represented to one selects just those input measurements that contain the significant data for tackling the specific issue. (Khalid, Khalil, & Nasreen, 2014)

**- Evaluation Metrics in Machine Learning:**

The evaluation metric can be portrayed as the measurement tool that measures the execution of the classifier.(Hossin, Sulaiman, 2015)

## 1.3 Problem Statement

With the huge advancement in machine learning and neural network, there must be an approach for these two approaches to contribute to the reduction of the number of deaths from breast cancer patients through early accurate prediction for the type of cancer.

There are also many ML algorithms that have been used to classify breast cancer whether it’s malignant or benign. Accordingly, many of such approaches are used such as MLP but with different accuracy rate. We need to carefully consider these differences in terms of accuracy when classifying breast cancer. As matter of fact, many ML algorithms differ in the way how to relate many attributes that shapes the tumours along with different schemas for weighting these attributes in order to relate or classify this cancer either malignant or benign.

## 1.4 Questions of the Study

This research is aimed to answer the following questions :

- Which machine learning algorithms (KNN, Random Forest & MLP) are best in breast cancer classification using specific dataset?
- Which attributes are more relevant to the classification of malignancy?
- How can we improve the accuracy of MLP to reach a higher result than the result of machine learning algorithms?

## 1.5 Goal and Objectives

The main objectives of this study are:

1. Comparing the ML algorithms of the classification on the WDBC dataset to obtain the best algorithms in the results.
2. Identify the most specific and relevant attributes of the malignant tumour classification through more than one feature selection algorithms.
3. Identify which activation function for MLP that produces a more accurate result.
4. Modification of special hyperparameters in MLP for best results.

## 1.6 Motivations

This research aims to find what is the best parameters for MLP as its basic deep learning approach in order to increase the accuracy of predicting breast cancer.

## 1.7 Contribution and Significance of the Research

This research aims to study, as a benchmark, a set of ML techniques in order to classify or predict breast cancer. With a plethora of techniques, each of which owns a process for building the predictive model. In addition to the importance of detecting the tumour in the early stages, the necessity to find sub-optimal models is required. This research presents the results of many ML methods that would be baselines for researchers to focus more on improving such techniques for the future of breast cancer diseases.

## 1.8 Scope of the Study

The scope of this thesis is inside the classification of breast cancer dependent on machine learning methodology. The work include examining upgrades of the MLP for accomplishing better accuracy, the work performs an investigation of open breast cancer datasets to gauge the planned extensions of machine learning.

## 1.9 limitations of the study

The study is comparative and limited to the use of WDBC dataset for machine learning algorithms for classification of breast cancer.

## 1.10 Thesis Outline

**This chapter** addresses the machine learning algorithms for classification of breast cancer in general and provides an overview of machine learning algorithms classification, the feature selection algorithms and machine learning evaluation metrics are also presented in this chapter. Finally, the research problem, the objectives, limitations, and scope are also discussed. The rest of this thesis is organized as follows:

**Chapter Two** examines and reviews the previous studies that are related to machine learning for classification of breast cancer. The related works are characterized and talked about extensively in a literature review. An overall literature review of strengths and weaknesses.

**Chapter Three** presents classification algorithms in machine learning in detail and feature selection algorithms. Analyze the classification algorithms in term of hyperparameters, dataset, and accuracy.

**Chapter Four** presents the implementation of the proposed MLP. The results and its effectiveness are also discussed in this chapter.

**Chapter Five** gives a general summary of the thesis, encapsulates the examination discoveries and future works.

# Chapter Two

# Background and Literature Review

# Chapter Two
# Background and Literature Review

## 2.1 Overview

This chapter presents the motivations and details of the approaches that have extended the Multi-Layer Perceptron (MLP). Section 2.2 presents an introduction to Machine learning. Section 2.3 explains the mechanism of Machine Learning Algorithm. Section 2.4 shows different previous studies associated with the ML algorithms used in this study for classification of breast cancer from 2015 until now. Section 2.5 presents the overall ML Classification comparison. Section 2.6 provides a summary of this chapter.

## 2.2 Introduction

Most modern ML models are based on an artificial neural network, although they can also include propositional formulas or latent variables organized layer-wise in deep generative models.

In ML, each level learns to transform its input data into a slightly more abstract and composite representation.

It also helps to disentangle these abstractions and pick out which features improve performance.

As for supervised learning tasks, ML methods obviate feature engineering, by translating the data into compact intermediate representations identical to principal components, and derive layered structures that remove redundancy in representation.

It can be also applied to unsupervised learning tasks. This is an important benefit because unlabeled data are more abundant than labelled data. (Schmidhuber, 2015).

## 2.3 Machine Learning Algorithm:

The classification techniques are most useful for gathering approaching instances dependent on some patterns and constraints. A sum total of 21 classification techniques of 9 distinct gatherings are mulled over for the evaluation of best classifiers and consequent prediction. In this study, we would devote top three out of thirty one technologies. Other techniques and their results in Appendix A.

### 2.3.1.1 Function Classifiers:

The classifiers under this group are no probabilistic in nature, where the system endeavors to sum up the training data before the real classification has occurred. Numerous variations of function classifiers have been proposed. (Dey et al., 2018)

- **Multi-layer Perceptron (MLP):** is a great spot to begin when you are learning about deep learning.

It is a supervised learning algorithm which comes under the functions classifier category that learns a function $f(.) = R^m \rightarrow R^o$ by training on a dataset, where ***m***, is the number of dimensions for input and ***o*** is the number of dimensions for output. Given a set of features $X = x_1, x_2, x_3, \ldots\ldots, x_m$ and a target ***y***, it can learn a non-linear function approximator for either classification or regression. It is different from logistic regression, there can be one or more non-linear layers, in that between the input and the output layer, called hidden layers. Figure 2.1 shows a one hidden layer MLP with scalar output.

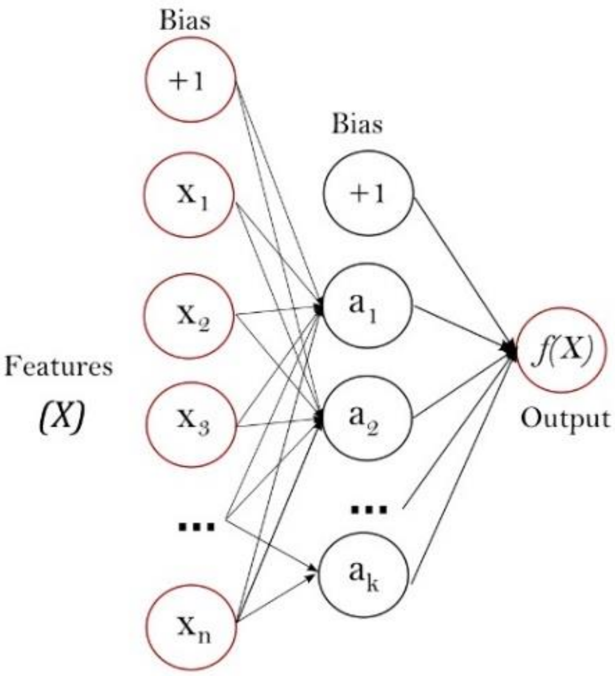



**Figure 2.1: One hidden layer MLP (scikit-learn.org, n.d.)**

**Advantages:** To use for regression and mapping, **t**he MLP algorithm is a generally very good algorithm. It tends to be utilized to map an N-dimensional input signal to an M-dimensional output signal, this mapping can likewise be non-linear.

**Disadvantages**: The principal restriction of the MLP algorithm is that the number of Hidden Neurons must be set by the user, setting this value too low may result in the MLP model underfitting while at the same time setting this value too high may result in the MLP model overfitting.

Another restriction of the MLP algorithm is that due to the manner in which it is trained, it can't ensure that the minima it stops at amid training are the global minima. The MLP algorithm can stall out in local minima. (NickGillian, 2014)

#### 2.3.1.2 Lazy Classifiers:

All the classifiers under this group are named as Lazy in light of the fact that as the name proposes speculation past the training data is deferred until an inquiry is made to the system.

That is, it doesn't construct a classifier until a new instance needs to be classified. Because of this reason, these classifiers are called instance based and consumes more computation time while building the model. The classifiers under the thought of lazy classifiers are KNN, Kstar, RseslibKnn, and privately weighted learning (LWL). (Dey et al., 2018)

- **K Nearest Neighbor (KNN):** which comes under the lazy classifier category, is a non-parametric supervised learning method in which we endeavor to classify the data point to a given classification with the assistance of the training set. In straightforward words, it catches information on all training cases and classifies new cases dependent on a likeness.

Predictions are made for a new instance (x) by searching through the entire training set for the K most similar cases (neighbors) and summarizing the output variable for those K cases. In classification, this is the mode (or most common) class value.

There are many distance functions however Euclidean is the most ordinarily utilized measure. It is mostly utilized when data is continuous. Manhattan distance is likewise basic for continuous variables.

Euclidean :

$$d(x, y) = \sqrt{\sum_{i=1}^{m} (x_i - y_i)^2}$$

Manhattan / city - block :

$$d(x, y) = \sum_{i=1}^{m} |x_i - y_i|$$

**Figure 2.2: Distance Functions (Deepanshu Bhalla, n.d.)**

**Advantages**: This algorithm is powerful to loud training data, easy to actualize, and compelling if training data is large.

**Disadvantages**: The calculation cost is high as it needs to a computer the distance of each instance to all the training tests, and need to decide the value of K (Garg, 2018)

**2.3.1.3 Trees Classifiers**:

The standard of part criteria is behind the intelligence of any decision tree classifier. Decision trees are displayed like a stream graph, with a tree structure wherein instances arc classified by their feature values. A node in a decision tree speaks to an instance. results of the test spoken to by the branch and the leaf node embodied the class label.

- **Random Forest**: As shown in Figure 2.3, Random Forest algorithm is a supervised classification algorithm. which comes under the trees classifier category, As the name proposes, this algorithm makes the forest with various trees. In general, the more trees in the forest the heartier the forest resembles. Similarly in the random forest classifier, the higher the quantity of trees in the forest gives the high accuracy results. (Mahajan & Ganpati, 2014)

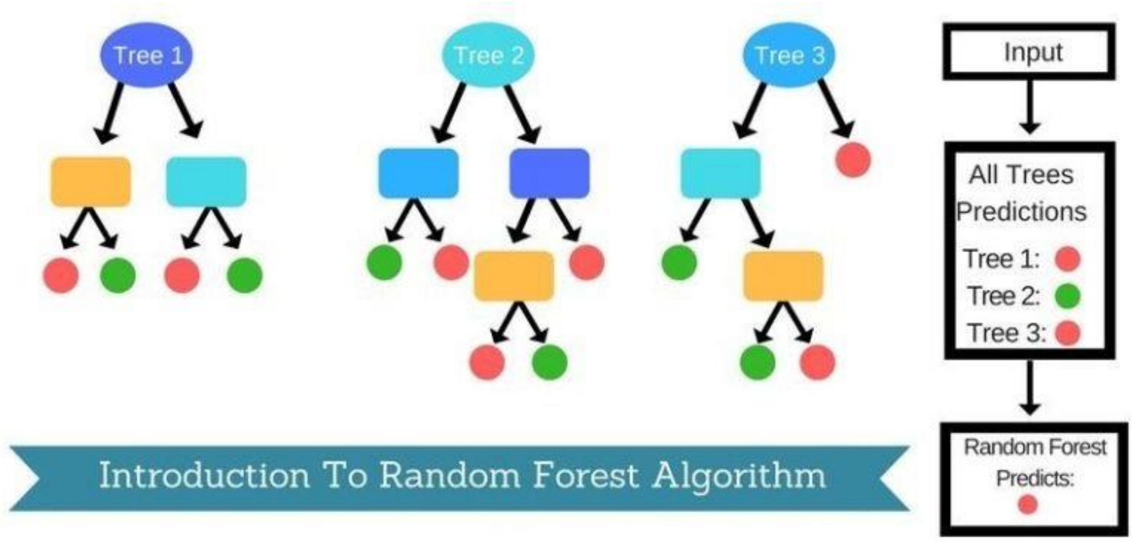



**Figure 2.3: Introduction to the random forest algorithm (Polamuri, 2017)**

**Advantages**: Reduction in over-fitting and random forest classifier is more precise than decision trees by and large.

**Disadvantages**: Slow real-time prediction, hard to implement, and complex algorithm. (Garg, 2018)

## 2.3.2 Feature Selection:

Because of a lot of data streaming over the network real-time machine learning is practically generally complex. Feature selection can reduce calculation time and model complexity. Research on feature selection began in the early 60s. Essentially, feature selection is a strategy of choosing a subset of pertinent/vital features by evacuating most irrelevant and redundant features from the data for structure a viable and proficient learning model. As shown in Figure 2.4.

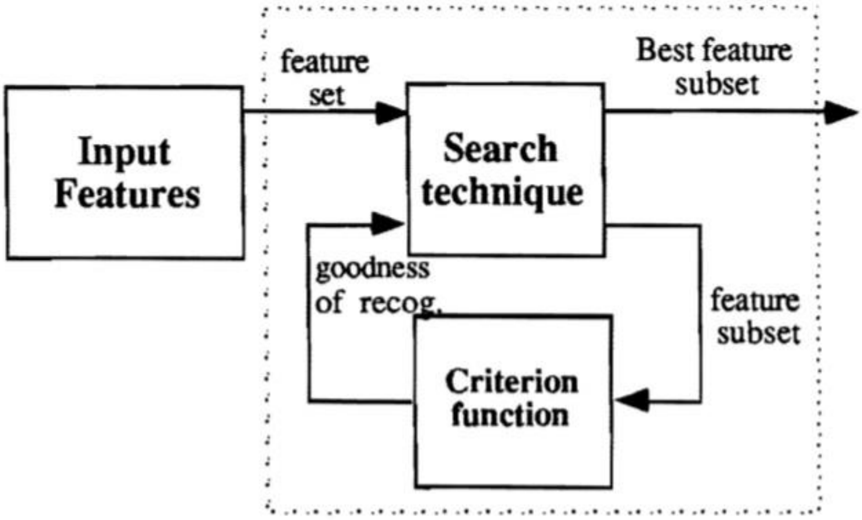



**Figure 2.4 Block diagram of the adaptive feature selection process (Vafaie & Imam, 1997)**

The procedure of Feature Selection forms, as shown in Figure 2.5, includes four fundamental strides in a run of the mill feature selection strategy appeared in Figure 1. First is age strategy to produce the following competitor subset; the second one is an assessment capacity to assess the subset and the third one is a halting rule to choose when to stop, and an approval technique to check whether the subset is valid. (Aggarwal, 2013)

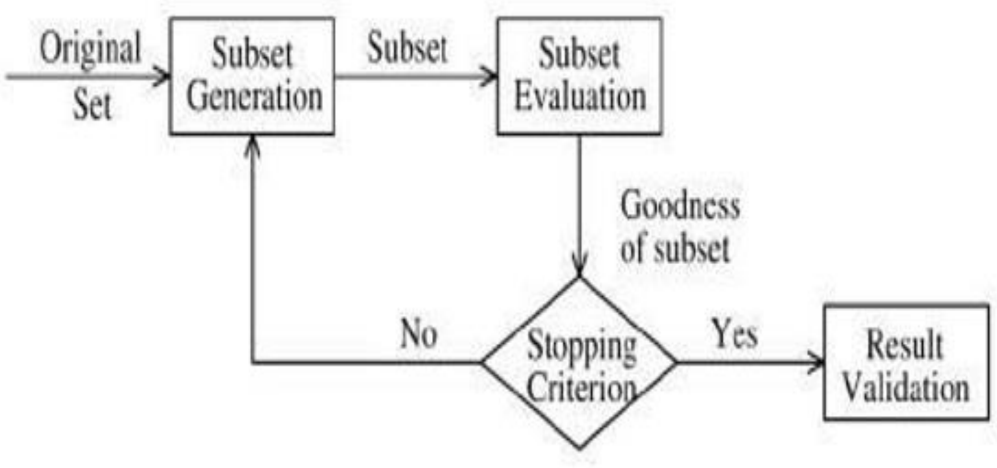



**Figure 2.5: Feature selection process (Aggarwal, 2013)**

**2.3.2.1 Attribute evaluators:**

**-InfoGainAttributeEval:** Evaluates the value of an attribute by estimating the information gain regarding the class.

**-CfsSubsetEva**l: Evaluates the value of a subset of attributes by considering the individual capacity of each feature alongside the level of redundancy between them. Subsets of features that are exceptionally related with the class while having low intercorrelation with different attributes are liked.

-**WrapperSubsetEval:** Evaluates attribute sets by using a learning scheme. Cross-validation is used to estimate the accuracy of the learning scheme for a set of attributes.

#### 2.3.2.2 Search Methods:

These methods search the arrangement of all possible features in order to find the best arrangement of features. Four search methods, which include (BestFirst, GeneticSearch, GreedyStepwise and Ranker).

**- Bestfirst:** This searches the space of attribute subsets by greedy hillclimbing increased with a backtracking facility. Setting the quantity of sequential non-improving nodes permitted controls the dimension of backtracking done.

**- GeneticSearch:** Genetic algorithms play out a search utilizing the straightforward genetic algorithm.

It typically maintains a constant-sized population of individuals which represent samples of the space to be searched. Each individual is evaluated on the basis of its overall fitness with respect to the given application domain. new individuals (samples of the search space) are produced by selecting high performing individuals to produce "offspring" which retain many of the features of their "parents". This eventually leads to a population that has improved fitness with respect to the given goal.

The main issues in applying GAs to any problem are selecting an appropriate representation and an adequate evaluation function. (Vafaie & Imam, 1997)

**- GreedyStepwise:** It plays out a greedy forward or backward search through the space of attribute subsets. May begin with no/all attributes or from a discretionary point in the space. Stops when the expansion/cancellation of any residual attributes results in a diminishing in the assessment. Can likewise deliver a positioned rundown of attributes by crossing the space from one side to the next and recording the request that attributes are selected.

**- Ranker:** It ranks attributes by their individual evaluations. Use in conjunction with attribute evaluators (Chisquare, GainRatio, InfoGainetc). (Aggarwal, 2013)

### 2.3.3 Confusion matrix

A confusion matrix demonstrates the number of correct and incorrect predictions made by the classification model with the genuine results (target value) in the data. The matrix is NxN, where N is the number of target values (classes). Execution of such models is normally evaluated utilizing the data in the matrix. The accompanying table 2.1 shows a 2x2 confusion matrix for two classes (Positive and Negative). (Sayad, 2011)

**Table 2.1 Confusion Matrix classes (Sayad, 2011)**

| Confusion Matrix | | **Target** | | | |
|---|---|---|---|---|---|
| | | Positive | Negative | | |
| **Model** | Positive | a | b | *Positive Predictive Value* | a/(a+b) |
| | Negative | c | d | *Negative Predictive Value* | d/(c+d) |
| | | *Sensitivity* | *Specificity* | **Accuracy** = (a+d)/(a+b+c+d) | |
| | | a/(a+c) | d/(b+d) | | |

- **Accuracy**: the proportion of the total number of predictions that were correct.
- **Positive Predictive Value or Precision:** the proportion of positive cases that were correctly identified.
- **Negative Predictive Value:** the proportion of negative cases that were correctly identified.
- **Sensitivity or Recall:** the proportion of actual positive cases which are correctly identified.
- **Specificity:** the proportion of actual negative cases which are correctly identified.

## 2.3.4 Evaluation metrics in Machine learning:

-**Accuracy (ACC):** As a rule, the accuracy metric measures the ratio of right predictions over the all-out number of cases evaluated. $acc = \frac{tp+tn}{tp+fp+tn+fn}$

-**Recall (r):** is utilized to measure the part of positive patterns that are accurately classified. $r = \frac{tp}{tp+tn}$

-**Receiver Operating Characteristic (ROC) curve**: is a graphical plot that delineates the diagnostic capacity of a binary classifier system as its segregation threshold is varied.

- **Area Under Curve (AUC):** is one of the mainstream ranking type metrics. the AUC was utilized to build an upgraded learning model and furthermore for looking at learning algorithms. In contrast to the limit and threshold metrics, the AUC value mirrors the general ranking execution of a classifier. For the two-class issue. $AUC = \frac{s_{p-n_p(n_{n+1})/2}}{n_p n_n}$

- **Precision (P):** is utilized to measure the positive patterns that are accurately anticipated from the absolute predicted patterns in a positive class. $p = \frac{tp}{tp+fp}$

- **F-Measure (FM):** This metric represents the consonant mean among recall and precision values. $FM = \frac{2 \times p \times r}{p+r}$

**Note:** each class of data; True positive (tp), False negative (fn), False positive (fp), True negative (tn), Specificity (sp), Sensitivity (sn), Precision (p).(Hossin & Sulaiman, 2015)

## 2.4 Related Works

In this section will illustrate a number of related works in order to determine the major research techniques and methodologies used. The following literature survey describes the previous work done on classification using ML researches. I would arrange the studies from the oldest to newest.

(Waugh et al., 2015): the authors conducted a survey Patient-tailored-made drugs for breast cancer patients that rely on histological and immunohistochemical (IHC) subtypes. Their findings were aimed at breast pathology. Subtype classifications were conducted by exploiting a cross-validated K-Nearest-Neighbor (KNN) (k=3) approach, with accuracy in respect to pathology assessed and receiver operator curve (AUROC) determined. Mann-Whitney U and Kruskal-Wallis experiments were utilized to assess crude entropy feature esteems. Their findings were as follows: Histological subtype classifications were comparable across training (n=148 cancers) and test sets (n=73 lesions) by all COM features (training: 75 %, AUROC=0.816; test: 72.5 %, AUROC=0.823). Entropy features were considerably diverse between lobular and ductal cancers ($p<0.001$; Mann Whitney U). Moreover, the IHC classifications using COM features were also akin for training and validation data (training: 57.2 %, AUROC=0.754; test: 57.0 %, AUROC=0.750) and their findings were that Textural differences on contrast-enhanced MR images may replicate underlying lesion subtypes, which merits testing against treatment response.

(Choi, 2015): the authors in this paper suggested the use of summed up multiple classifier systems enhance the classification of mammographic masses in Computer-aided detection (CAD). Their approach aimed to invigorate different base (component) classifiers to learn distinctive pieces of an article moment space, they generated a classifier based systems that combine data resampling underpinning AdaBoost along with the exploitation of different feature. Moreover, their classifier systems can be summed up past the confinement of powerless classifiers in ordinary AdaBoost learning.

(Diz, Marreiros, & Freitas, 2016): in this paper, the authors compared Random forest and Naive Bayes classifiers on two different breast cancer datasets and to the best methods in predicting benign/malignant lesions, breast density classification. Their comparison was based on two matrices of texture features extraction, and their findings were that Naive Bayes was the best to identify masses texture, and Random Forests was the first- or second-best classifier for the majority of tested groups.

(Sahu & Miri, 2017): in this paper, the authors proposed a hybrid technique by combining random committee algorithm and voted Perceptron algorithm in order to minimize the error rate for predicting the Brest cancer. Their approach relies on voting to select the features from the dataset that directly relate to the Brest cancer that also led to the reduction of error rate when predicting the appearance of Brest cancer.

(Nilashi, Ibrahim, Ahmadi, & Shahmoradi, 2017): the authors proposed a knowledge-based system for predicting breast cancer. Their system used Expectation Maximization (EM) and Principal Component Analysis (PCA) were individually exploited for clustering and addressing the multicollinearity problem within the datasets. Moreover, they used Regression Trees (CART) to consequently create a set of fuzzy rules from the dataset that will help predict the presence of Brest cancer of mammogram images. they evaluated the knowledge-based system with respect to two real-world datasets, WDBC and Mammographic mass.

(Yue, Wang, Chen, Payne, & Liu, 2018): they presented a comparative study of using artificial neural networks (ANNs), support vector machines (SVMs), decision trees (DTs), and k-nearest neighbours (k-NNs) to classify Brest cancer and determine which classifier was more accurate, their experiments were conducted using the WBCD dataset.

(MOHAN & NAGARAJAN, 2019): To improve the classification, this investigation utilizes an ensemble based feature selection utilizing random trees and wrapper technique. The proposed ensemble learning classification technique infers a subset utilizing the wrapper strategy, bagging, and random trees. The proposed strategy evacuates the unimportant features and selects the ideal features for classification through probability weighting criteria. The improved algorithm can recognize the important features from immaterial features and improve the classification performance. The proposed features selection technique is evaluated utilizing SVM, RF, and NB evaluators and the exhibitions are thought about against the FSNBb, FSSVMb, GASVMb, GANBb, and GARFb strategies. The proposed strategy accomplishes mean classification accuracy of 92% and beats the other ensemble techniques.

(Li, Gao, & D'Agostino, 2019): they provided a tutorial for evaluating classification accuracy for various state-of-the-art learning approaches, including familiar shallow and deep learning methods. For qualitative response variables with more than two categories, many traditional accuracy measures such as sensitivity, specificity, and area under the receiver operating characteristic curve are not applicable and they had to consider their extensions properly. They considered a few important statistical concepts for multicategory classification accuracy were reviewed and their utilities for various learning algorithms were demonstrated with real medical examples. they offered a problem-based R code to illustrate how to perform these statistical computations step by step. they expected that such analysis tools will become more familiar to practitioners and receive broader applications in biostatistics.

## 2.5 ML Classification Comparison

After reviewing previous studies that are related to ML classification, a comparison between these variations is conducted as given in Table 2.2.

**Table 2.2: Comparison between ML Classifications**

| ID | Title | Year | The main goal | Strength Points | Weakness Points |
|---|---|---|---|---|---|
| 1 | Magnetic resonance imaging texture analysis classification of primary breast cancer | 2015 | Patient-tailored-made medications for breast cancer depend on histological and immunohistochemical (IHC) subtypes. Magnetic Resonance Imaging (MRI) texture analysis (TA), Investigations were blinded to breast pathology. | • utilizing a cross-validated K-Nearest-Neighbor (KNN) (k=3) strategy, with accuracy in respect to pathology assessed and receiver operator curve (AUROC) determined | • Used histological and immunohistochemical (IHC) subtypes. Magnetic Resonance Imaging (MRI) texture analysis |
| 2 | A Generalized Multiple Classifier System for Improving Computer-aided Classification of Breast Masses in Mammography | 2015 | display a summed up multiple classifier systems for the improved arrangement of mammographic masses in Computer-aided detection (CAD). | • proposed multiple classifier systems<br>• The used area under the receiver operating characteristic (AUC) and the normalized partial area under the curve (pAUC). | • Used mammographic masses<br>• single neural network (NN) and support vector machine (SVM) based characterization approaches. |
| 3 | Applying Data Mining Techniques to Improve Breast Cancer Diagnosis | 2016 | compare two breast cancer datasets and find the best methods in predicting benign/malignant lesions, breast density classification, and even for finding identification (mass/microcalcification distinction). | • Used Random Forests RF | • Used Naive Bayes NB<br>• Using Matlab<br>• Breast Cancer Digital Repository (BCDR-D01 (Digital Mammography dataset number 1) was used, with 64 mammograms of women |
| 4 | A Hybrid Technique for creating classification model using Random Committee and Voted Perceptron Classifier | 2017 | minimize the error rate for Classification so that they could reach the nearest possible to correct classification. | • Used voted perceptron algorithm | • Used random committee algorithm<br>• Using dataset from 435 instances and 15 attributes. |
| 5 | A knowledge-based system for breast cancer classification using fuzzy logic method | 2017 | proposed another knowledge-based system for breast cancer disease using clustering, noise removal, and classification techniques | • Using two datasets, one of them: WDBC<br>• Using classification techniques | • Using two datasets, one of them: Mammographic mass.<br>• Using clustering, noise removal |
| 6 | Machine Learning with Applications in Breast Cancer Diagnosis and Prognosis | 2018 | overview of ML techniques including artificial neural networks (ANNs), support vector machines (SVMs), decision trees (DTs), and k-nearest neighbours (k-NNs). | • Using k-nearest neighbours (k-NNs).<br>• Used WBCD. | • Using ANN, SVM, DT |
| 7 | An improved tree model based on ensemble feature selection for classification | 2019 | ensemble based feature selection utilizing random trees and wrapper technique to improve the classification | • proposed features selection technique is evaluated utilizing SVM, RF, and NB evaluators and the exhibitions are thought about against the FSNBb, FSSVMb, GASVMb, GANBb, and GARFb strategies | • fifteen different datasets were taken from the UCI repository |
| 8 | Evaluating classification accuracy for modern learning approaches | 2019 | tutorial for evaluating classification accuracy for various state-of-the-art learning approaches, including familiar shallow and deep learning methods | • Using multilayer perceptron (MLP) | • R code<br>• convolutional neural network (CNN) |

## 2.6 Summary

In this chapter, several studies that are related to ML classification for breast cancer were reviewed, described, analyzed, and summarized. This review included the advantages, limitations, objectives, and effectiveness of the ML classification for breast cancer. In general, the existing approaches aim at either increasing the performance of the original MLP or reducing attributes for instances size. The MLP extensions resulted in a classifier that is comparable to the original MLP.

Each of the discussed literature in this chapter achieved a certain goal, but as noted, literature is lacking an MLP, closer that deep learning, that combines the discriminative power, increasing performance and reducing the size of the vector feature. Thus, there is a need to tuning an MLP with such properties.

# Chapter Three
# Methodology and the Proposed Work

# Chapter Three
# Methodology and the Proposed Work

## 3.1 Overview

This chapter presents the proposed work, the organization of this chapter is as follows: Section 3.2 introduces the chapter behind tuning the MLP classifier. Section 3.3 presents and analyzes the proposed tuned MLP methodology and its implementation steps. Section 3.4 shows the flowchart constructs an algorithm. Finally, Section 3.5 gives a summary of this chapter.

## 3.2 Introduction

The classification is a core process for a large number of breast cancer prediction, which requires features that are extracted from dataset, in addition to an algorithm to implements the classification task. Recently, reducing features from a dataset, such as MLP feature, were discussed extensively in the literature. As mentioned in Chapter Two, Tuned MLP extensions are generally divided into two categories, one investigated the reduction of the extracted feature size, while, the other investigated the enhancement of the classification power. This chapter introduces a Tuned approach for local feature extraction by extending the MLP with the aim of reducing the size of the extracted feature and enhance the classification power discriminating different ML classification algorithms.

## 3.3. The Tuned MLP Methodology

The methodology used in this thesis as illustrated in figure 3.1, consist of the following steps:

*A. Data Selection and Preparation*

In this work, the dataset is selected from WDBC which is in CSV file format. This dataset includes 569 instances and 32 attributes.

*B. Tool Selection*

Keras-scikit and Weka used in this work which supports various data mining tasks and it also includes a collection of different classifiers.

*C. Preprocessing Data*

Now the data set is loaded then a supervised filter which is attribute based is applied on the data set which removes unnecessary attributes and therefore 1 attribute was selected.

*D. Classification Iteration 1*

Then we determine to classify and select cross validation 10 and select the Random Forest algorithm.

*E. Classification Iteration 2*

Then we determine to classify and select cross validation 10 and select the KNN algorithm.

*F. Classification Iteration* 3

Then we determine to classify and select cross validation 10 and select the MLP algorithm.

*G. Ensemble feature selection*

Then go to select attributes and choose a search method: Ranker and from attribute evaluators select InfoGainAttributeEval. Then apply the same steps: search method (BestFirst) with attribute evaluators (CfsSubsetEval), search method (Greedy Stepwise) with attribute evaluators (CfsSubsetEval), search method (BestFirst) with attribute evaluators (WrapperSubsetEval), and search method (Greedy Stepwise) with attribute evaluators (WrapperSubsetEval)

*H. Voted of the best relevant features*

we voted of the best relevant features through the following equation (3.1) :

$$Y = W_1C_1+W_2C_2 + W_3C_3+W_4C_4+ W_5C_5 \tag{3.1}$$

*I. Classification using the tuned MLP*

Keras-scikit used in this work which supports various data mining tasks and it also includes a collection of different classifiers. By trying and experimenting and comparing the parameters of the MLP algorithm to raise its accuracy in the classification of breast cancer.

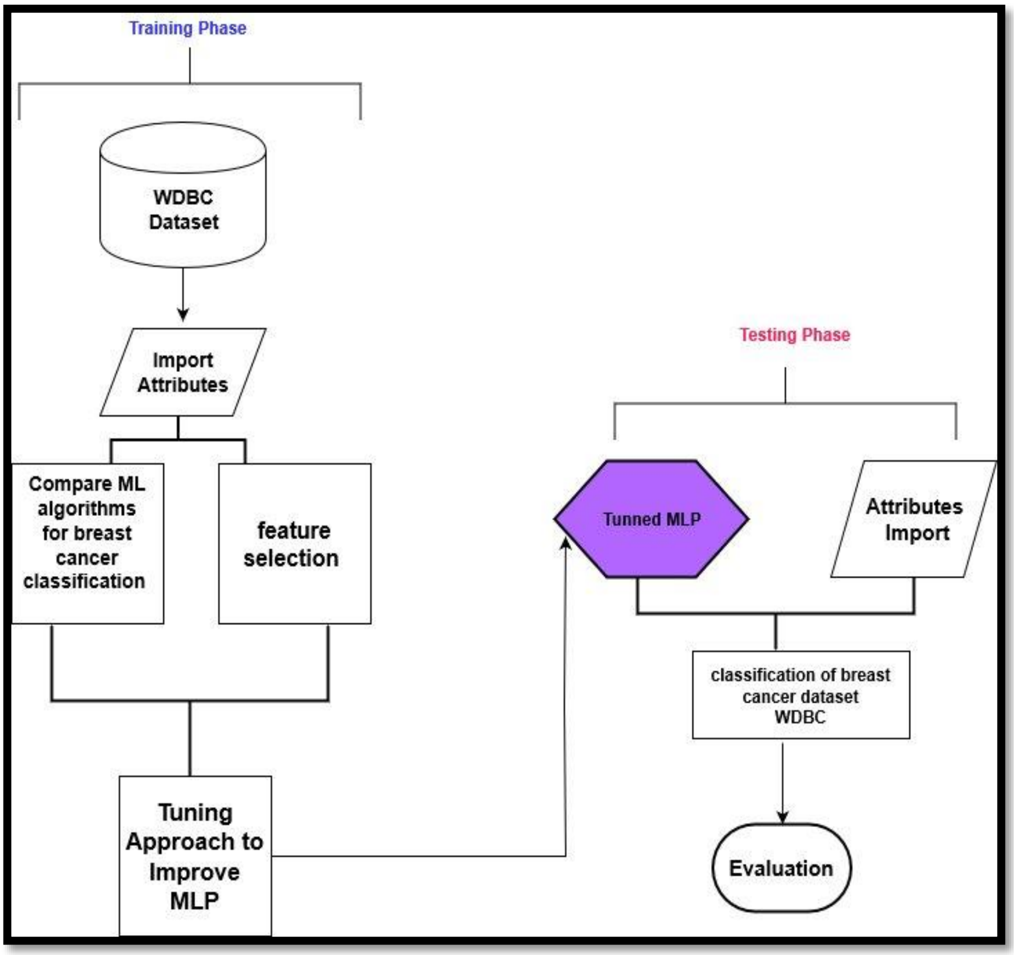



**Figure 3.1: The Tuned MLP**

## 3.4 Summary

In this chapter, an MLP-extension, called Tuned MLP was proposed. used Keras-scikit and Weka in this work which supports various data mining tasks and it also includes a collection of different classifiers. Then we go to classify and use cross-validation 10 and select the RF, KNN and MLP algorithms. Then go to select attributes: (search method with attribute evaluators) next voted of the best relevant features, finally, classification using the tuned MLP by trying and experimenting and comparing the parameters of the MLP algorithm to raise its accuracy in the classification of breast cancer.

# Chapter Four

# Implementation and Results

# Chapter Four
# Implementation and Results

## 4.1 Overview

This chapter shows the experimental results of the proposed MLP. This chapter is sorted out as pursues: Section 4.2 gives an introduction to the investigations led. Section 4.3 talks about the implementation of subtleties. Section 4.4 presents the parameter settings for the proposed and compared approaches. Section 4.5 presents the measurements that are utilized to assess the proposed MLP. Section 4.6 presents the results of the implementation and demonstrates the performance of the proposed MLP. At long last, Section 4.7 gives a summary for this chapter.

## 4.2 Introduction

The proposed MLP, which was presented in Chapter 3, is implemented using Python programming language. The implementation stages, to obtain and compare the results, are presented in this chapter. Besides the proposed MLP, other ML classification algorithms, which are KNN and Random forest, will be evaluated and compared with the proposed MLP. The performance evaluation of the proposed and compared ML classification will be tested using the WDBC dataset. Accuracy measure is used for performance rating and compares outcomes.

## 4.3 Dataset

This section describes the properties and lists some statistics about the utilized dataset.

### 4.3.1 WDBC Dataset

Features are figured from a digitized image of a fine needle aspirate (FNA) of a breast mass. They depict the characteristics of the cell cores present in the image. n the 3-dimensional space is that portrayed in: [K. P. Bennett and O. L. Mangasarian: "Robust Linear Programming Discrimination of Two Linearly Inseparable Sets", Optimization Methods and Software 1, 1992, 23-34].

Attribute Information:

1) ID number 2) Diagnosis (M = malignant, B = benign) 3-32)

Ten real-valued features are computed for each cell nucleus:

a) radius (mean of distances from center to points on the perimeter)

b) texture (standard deviation of gray-scale values)

c) perimeter

d) area

e) smoothness (local variation in radius lengths)

f) compactness (perimeter^2 / area - 1.0)

g) concavity (severity of concave portions of the contour)

h) concave points (number of concave portions of the contour)

i) symmetry

j) fractal dimension ("coastline approximation" - 1)

The mean, standard error and "worst" or largest (mean of the three largest values) of these features were computed for each image, resulting in 30 features. For instance, field 3 is Mean Radius, field 13 is Radius SE, field 23 is Worst Radius.

All feature values are recorded with four significant digits. Missing attribute values: none Class distribution: 357 benign, 212 malignant. (Wolberg, WIlliam H. Mangasarian, 2016) .Appendix B for more details.

## 4.4 Implementation

The proposed and looked at tuned MLP, other than the ML classification strategy, which is utilized for comparison purpose, are implemented utilizing python programming language. The figure 4.1 explain pseudo-code for the tuned MLP procedure.

```
Array1= gridsearch ()
Aray2= Manual_parameters
Final_array=Array1+Array2
While ( not_stopping):
            Tunning_model:
                  For I in layer:
                        Tunning_model(add_layer)
                  Tunning_model.classify(final_array)
Not stopping:
If val(accuracy) is fixed or decrease
Manual_parameters:
Activation1= [relu, tanh, sigmoid]
Optimizer=[rmsprop,adem]
gridsearch () : the package in Python, Searches for the best values for parameters and
applies them to the MLP algorithm and the WDBC dataset. Parameters are (seed,
kernel_initializer, dropout, loss, epochs, batch_size, lr)
```

**Figure 4.1: Pseudo-code for the tuned MLP procedure**

## 4.5 Parameter Setting

The implementation needs a collection of parameters that require setting for WDBC dataset. The same settings are applied to the proposed and compared ML classification, as listed in Table 4.1.

**Table 4.1: Parameter Settings of ML classification algorithms**

| Parameter | KNN | RF | MLP |
|---|---|---|---|
| **Input layer** | 31 | 31 | 31 |
| **Output layer** | 1 | 1 | 1 |
| **Type of class** | binary | binary | binary |
| **dataset** | WDBC | WDBC | WDBC |

The Normalize histogram is used to convert the integer values to the range of [0~1], by dividing each histogram bin value over the total sum of all bins. Finally, each instance will have equal size feature vector that will be inserted into the Tuned MLP for classification.

The tuning of hyperparameters for improving the result accuracy of MLP, this is where the magic happens, and there are as shown in table 4.2:

**Table 4.2: Hyperparameters Settings of Tuned MLP**

| Parameter | Value or Relation |
|---|---|
| **Number of hidden layers** | 6 |
| **Epochs** | 150 |
| **Optimizer** | Adem |
| **Batch size** | 5 |
| **Activation Functions** | sigmoid, sigmoid |
| **Learning rate** | 0.001 |
| **Seed** | 2 |

**Epochs**: is a hyperparameter that defines the number times that the learning algorithm will work through the entire training dataset. One epoch means that each sample in the training dataset has had an opportunity to update the internal model parameters. An epoch is comprised of one or more batches. (Brownlee, 2018)

**Optimizer**: Optimizers update the weight parameters to minimize the loss function. Loss function acts as guides to the terrain telling optimizer if it is moving in the right direction to reach the bottom of the valley, the global minimum.(Khandelwal, 2019)

**Batch size:** is a hyperparameter that defines the number of samples to work through before updating the internal model parameters. From this error, the update algorithm is used to improve the model, e.g. move down along the error gradient. A training dataset can be divided into one or more batches. (Brownlee, 2018)

**Activation Functions**: The activation functions also known as transfer function are typically a non-linear function that transforms the weighted sum of the inputs (the internally generated sum) to an output value. Sometimes different activation functions [4-14] are acquired for different networks so that it resulting in better performances. An activation function or transfer functions for the hidden nodes in MLP are needed to introduce nonlinearity into the network. (Isa et al., 2010)

**Learning rate:** a hyperparameter that controls how much to change the model in response to the estimated error each time the model weights are updated. (Brownlee, 2019).

**Seed**: is a collection of information that is used as training, testing, or as a template. For example, out computer dictionary terms could be used as seed data for anyone interested in writing their own version of a computer dictionary or who needed ideas of computer terms. (Computer Hope, 2017)

## 4.6 Evaluation Metrics

**Classification Accuracy**, which is referred to as the capacity of the algorithm to anticipate the correct class name for instances of obscure class labels (testing set), is determined as given in equation 4.1. Accuracy measure is utilized for evaluating and comparing the underlying ML classification.

$$\text{Classification Accuracy} = \frac{Number\ of\ correctly\ classified\ samples}{Total\ number\ of\ test\ samples} \times 100 \tag{4.1}$$

## 4.7 Results

In this section, the results of the proposed tuned MLP and the other comparison methods (KNN and RF) are presented. The results were categorized into two classes:

1. The results of KNN, Random Forest, the basic MLP without additional modification, which is referred to as basic mode.
2. The results of voting of features selection for search methods with attribute evaluators.
3. The results of tuning MLP, by implementing extra hyperparameters for the proposed and compared evaluation to raise the accuracy of classification.

### 4.7.1 Results of ML algorithms for the classification on the WDBC dataset

Table 4.3 lists the values of the classification accuracy of ML algorithms. As noted, the KNN and RF give the best results, whereas, the original MLP is the second best.

**Table 4.3: Accuracy results of ML algorithms for the classification on the WDBC dataset before selected attributes**

| Scheme | Accuracy | | | | | | |
|---|---|---|---|---|---|---|---|
| | Correctly Classified Instances | Incorrectly Classified Instances | Kappa statistic | Mean absolute error | Root mean squared error | Relative absolute error | Root relative squared error |
| **KNN** | 100.00% | 0.00% | 1.00 | 0.00 | 0.00 | 0.37% | 0.36% |
| **RF** | 100.00% | 0.00% | 1.00 | 0.03 | 0.07 | 5.81% | 13.55% |
| **MLP** | 96.66% | 3.34% | 0.93 | 0.04 | 0.17 | 7.82% | 34.98% |

Table 4.4 lists the values of the classification evaluation of ML algorithms. As noted, the KNN and RF give the best results, whereas, the original MLP is the second best.

**Table 4.4: Evaluation metrics results of ML algorithms for the classification on the WDBC dataset before selected attributes**

| Scheme | Evaluation Metrics | | | |
|---|---|---|---|---|
| | **Precision** | **Recall** | **F-Measure** | **ROC Area** |
| **KNN** | 1.00 | 1.00 | 1.00 | 1.00 |
| **RF** | 1.00 | 1.00 | 1.00 | 1.00 |
| **MLP** | 0.97 | 0.94 | 0.95 | 0.99 |

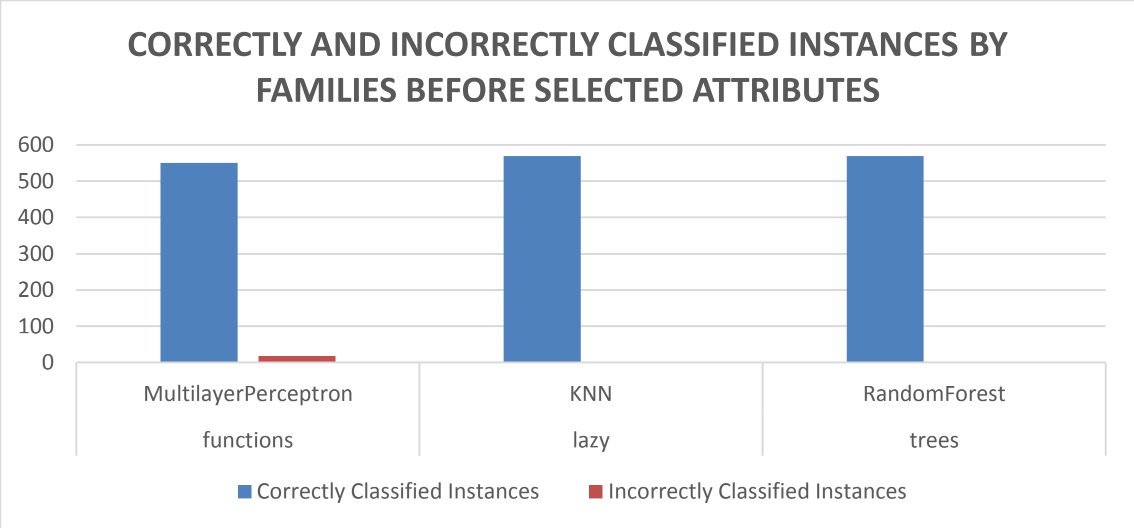



**Figure 4.2: illustrates the classification accuracy of the compared methods over WDBC Dataset in Basic mode and before selected attributes mode.**

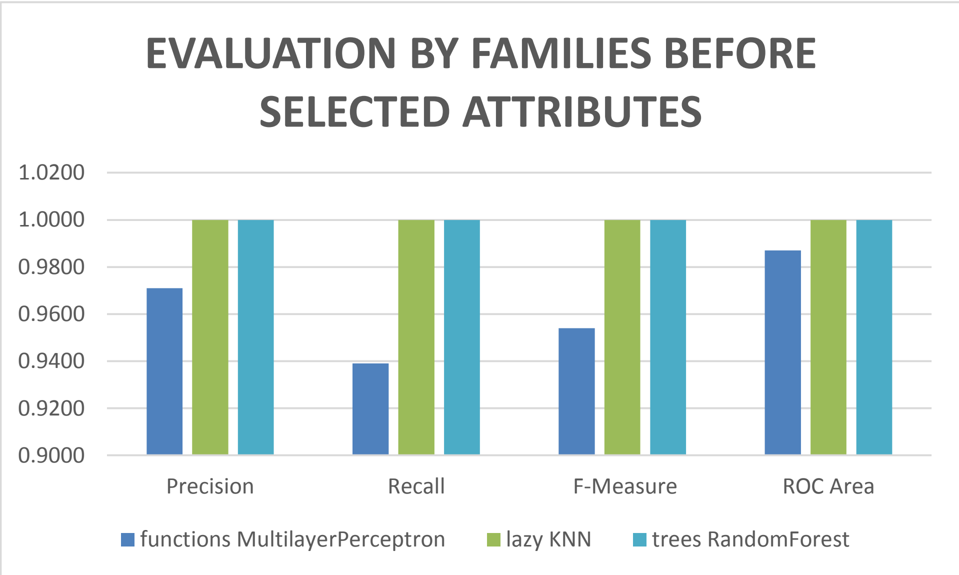



**Figure 4.3: Evaluation metrics results of ML algorithms for the classification on the WDBC dataset before selected attributes**

As noted in Figure 4.2 and Figure 4.3, the KNN and RF have obtained better accuracy compared to the original MLP in the basic mode. The best result for ML classification was for KNN and RF with an accuracy of 100%, the original MLP with an accuracy of 96.66%. The evaluation results of KNN and RF are (1,1,1,1) for (Precision, Recall, F-Measure, ROC Area) respectively and the original MLP is (0.97, 0.94, 0.95, 0.99) same order as before.

### 4.7.2 Results of voting of features selection for search methods with attribute evaluators

Table 4.5 lists the values of the classification accuracy of ML algorithms after selected attributes. As noted, the KNN and RF give the best results, whereas, the original MLP is the second best.

**Table 4.5: Accuracy results of ML algorithms for the classification on the WDBC dataset after selected attributes**

| Scheme | Accuracy | | | | | | |
|---|---|---|---|---|---|---|---|
| | Correctly Classified Instances | Incorrectly Classified Instances | Kappa statistic | Mean absolute error | Root mean squared error | Relative absolute error | Root relative squared error |
| **KNN** | 100.00% | 0.00% | 1.00 | 0.00 | 0.00 | 0.37% | 0.36% |
| **RF** | 100.00% | 0.00% | 1.00 | 0.03 | 0.07 | 5.81% | 13.55% |
| **MLP** | 97.19% | 2.81% | 0.94 | 0.04 | 0.15 | 7.80% | 31.03% |

Table 4.6 lists the values of the classification evaluation of ML algorithms after selected attributes. As noted, the KNN and RF give the best results, whereas, the original MLP is the second best.

**Table 4.6: Evaluation metrics results of ML algorithms for the classification on the WDBC dataset after selected attributes**

| Scheme: | Evaluation Metrics | | | |
|---|---|---|---|---|
| | **Precision** | **Recall** | **F-Measure** | **ROC Area** |
| **KNN** | 1.00 | 1.00 | 1.00 | 1.00 |
| **RF** | 1.00 | 1.00 | 1.00 | 1.00 |
| **MLP** | 0.97 | 0.95 | 0.96 | 0.99 |

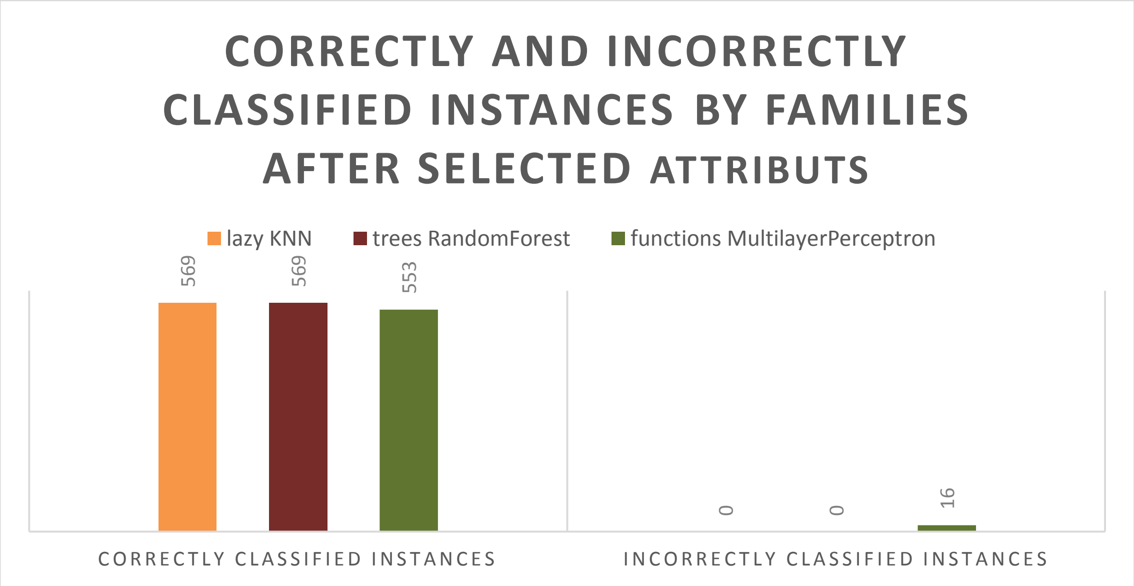



**Figure 4.4: illustrates the accuracy results of ML algorithms for the classification on the WDBC dataset after selected attributes.**

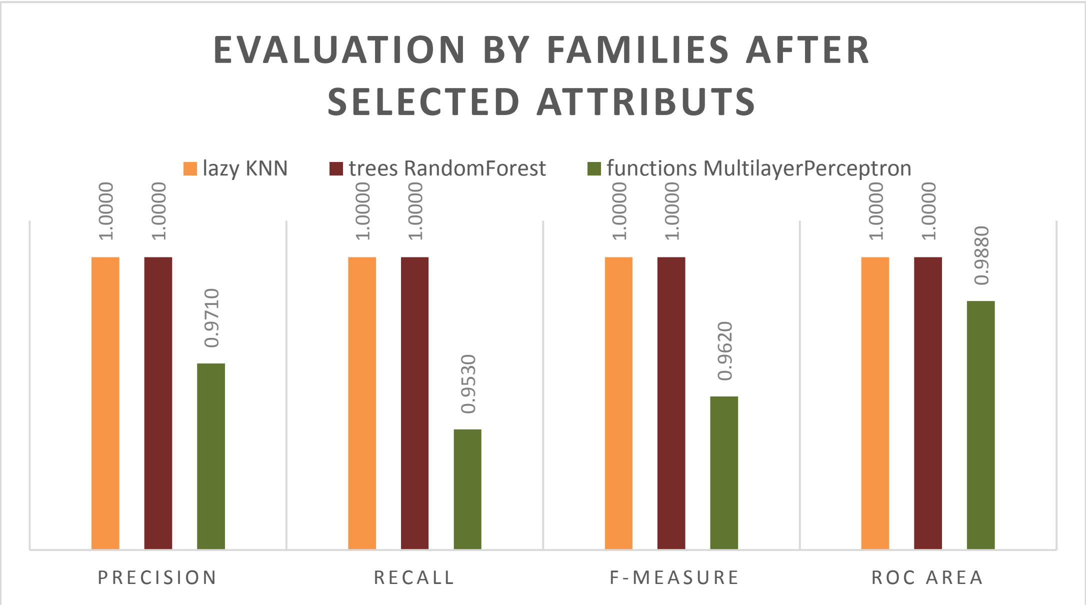



**Figure 4.5: Evaluation metrics results of ML algorithms for the classification on the WDBC dataset after selected attributes**

As noted in Figure 4.4 and Figure 4.5, the results of classification have obtained better accuracy compared to the same ML algorithms before selected attributes. The best result for breast cancer classification was for KNN and RF with an accuracy of 100.00%, the second rank for the original MLP with an accuracy of 97.19%.

The main reason for these differences was to reduce the feature selection from 31 attributes to only 4 attributes that were most influential by identifying the tumour type. Search methods were implemented with different attribute evaluators, for example, Ranker with InfoGain AttributeEval, Best First with CfsSubsetEval, Greedy Stepwise with CfsSubsetEval, Best First with Wrapper SubsetEval, and Greedy Stepwise Wrapper SubsetEval, we voted of the best relevant features through the following equation:

$$\mathbf{Y = w_1c_1+w_2c_2 + w_3c_3+w_4c_4+ w_5c_5}$$

The number of attributes was reduced from 31 to 4 attributes results indicated four attributes : ( **texture_mean, concave points_mean, Radius_worst, Smoothness_worst)**

### 4.7.3 Results of tuning MLP

The results of tuning MLP, by implementing extra hyperparameters for the proposed and compared evaluation to raise the accuracy of classification.

The tuned MLP is mainly about the best fit for MLP in order to improve accuracy, where a number of parameters that affect the MLP result that we studied and experimented in this thesis. That parameters are as follows, N: the number of the hidden layers, activation function for the first N-1 of the hidden layer, the last hidden layer activation function, the optimizer, hyperparameter (seed, learning rate, patch size, number of epochs, dropout to avoid overfitting problem ). The values we considered in our experiments are illustrated in Table 4.7

**Table 4.7: Tuning MLP Parameters**

| *hidden layer* | *activation1* | *optimizer* | Hyperparameters | acc |
|---|---|---|---|---|
| 1 | relu | rmsprop | rmsprop relu 1 hidden layers | 0.974 |
| 1 | relu | adem | adem relu 1 hidden layers | 0.961 |
| 1 | tanh | rmsprop | rmsprop tanh 1 hidden layers | 0.967 |
| 1 | tanh | adem | adem tanh 1 hidden layers | 0.967 |
| 1 | sigmoid | rmsprop | rmsprop sigmoid 1 hidden layers | 0.965 |
| 1 | sigmoid | adem | adem sigmoid 1 hidden layers | 0.970 |
| 2 | relu | rmsprop | rmsprop relu 2 hidden layers | 0.975 |
| 2 | relu | adem | adem relu 2 hidden layers | 0.968 |
| 2 | tanh | rmsprop | rmsprop tanh 2 hidden layers | 0.972 |
| 2 | tanh | adem | adem tanh 2 hidden layers | 0.970 |
| 2 | sigmoid | rmsprop | rmsprop sigmoid 2 hidden layers | 0.972 |
| 2 | sigmoid | adem | adem sigmoid 2 hidden layers | 0.970 |
| 4 | relu | rmsprop | rmsprop relu 4 hidden layers | 0.963 |
| 4 | relu | adem | adem relu 4 hidden layers | 0.970 |
| 4 | tanh | rmsprop | rmsprop tanh 4 hidden layers | 0.956 |
| 4 | tanh | adem | adem tanh 4 hidden layers | 0.967 |
| 4 | sigmoid | rmsprop | rmsprop sigmoid 4 hidden layers | 0.975 |
| 4 | sigmoid | adem | adem sigmoid 4 hidden layers | 0.970 |
| 6 | relu | rmsprop | rmsprop relu 6 hidden layers | 0.963 |
| 6 | relu | adem | adem relu 6 hidden layers | 0.967 |
| 6 | tanh | rmsprop | rmsprop tanh 6 hidden layers | 0.968 |
| 6 | tanh | adem | adem tanh 6 hidden layers | 0.963 |
| 6 | sigmoid | rmsprop | rmsprop sigmoid 6 hidden layers | 0.972 |
| 6 | sigmoid | adem | adem sigmoid 6 hidden layers | 0.977 |
| 8 | relu | rmsprop | rmsprop relu 8 hidden layers | 0.959 |
| 8 | relu | adem | adem relu 8 hidden layers | 0.970 |
| 8 | tanh | rmsprop | rmsprop tanh 8 hidden layers | 0.940 |
| 8 | tanh | adem | adem tanh 8 hidden layers | 0.656 |
| 8 | sigmoid | rmsprop | rmsprop sigmoid 8 hidden layers | 0.921 |
| 8 | sigmoid | adem | adem sigmoid 8 hidden layers | 0.695 |
| 10 | relu | rmsprop | rmsprop relu 10 hidden layers | 0.954 |
| 10 | relu | adem | adem relu 10 hidden layers | 0.961 |
| 10 | tanh | rmsprop | rmsprop tanh 10 hidden layers | 0.963 |
| 10 | tanh | adem | adem tanh 10 hidden layers | 0.629 |
| 10 | sigmoid | rmsprop | rmsprop sigmoid 10 hidden layers | 0.627 |
| 10 | sigmoid | adem | adem sigmoid 10 hidden layers | 0.627 |

As shown in table 4.7, the experiments were conducted by exploiting the use of Grid Search to find the best of the value of parameters of optimizers in terms of epoch, batch-size, seed. The result of experiments is evaluated based on which of the Grid Search produces the best result in terms of accuracy as shown in figure 4.6. Appendix C for more figures.

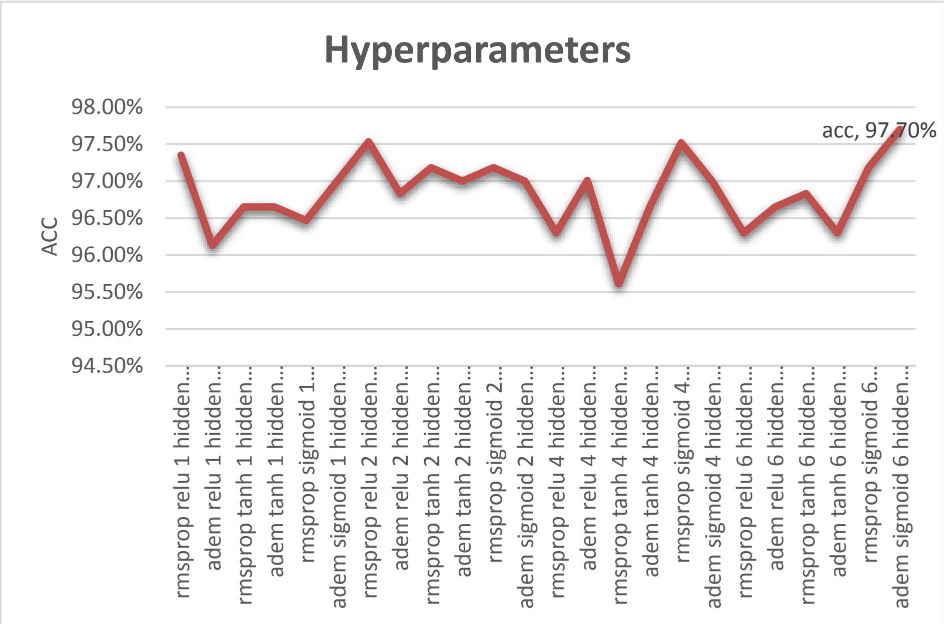



**Figure 4.6: Experiments Tuning MLP Parameters**

As noted in Figure 4.6, the best parameters that produces accuracy (97.70%), which is better as compared to the basic MLP (96.66%) as discussed early in this chapter. The best fit for the Tuned MLP as proposed in this thesis are as follows: as the activation function for the first N-1 Hidden layers is sigmoid, the number of hidden layers is six, the optimizer is Adem, accuracy

## 4.8 Summary

This chapter presents the implementation of the tuned MLP using python programming language over WDBC dataset to evaluate the classification accuracy of the tuned MLP. Two different ML classification algorithms, which are KNN and RF are used for comparison purposes. The empirical results of the tuned MLP demonstrate that its accuracy is satisfying in breast cancer classification.

In WDBC dataset, the results were the best rate accuracy for the KNN and RF classification algorithms. We also summarized the number of attributes from 31 to 4 attributes, which are most influential in determining the type of tumour (benign or malignant), where the accuracy of the classification increased. Lastly, we tuned MLP and its accuracy rate of 1.07 % compared to the original MLP.

# Chapter Five

# Conclusion and Future Work

# Chapter Five
# Conclusion and Future Work

## 5.1 Conclusion

In this thesis, presented a set of proposed methods for classify breast cancer disease. In particular, we conducted extension experiments on a well-known dataset using ML techniques such as MLP, RF, KNN.

The MLP technique was selected as method of study on which we updated its parameters in order to reach a reasonable classifier for breast cancer disease. The result showed 1.07% enhancement of accuracy 97.70% on six hidden layers and other optimal parameters. In addition, we exploited the feature selection techniques to select the most relevant attributes of high impact on learning technique.

The results showed with five features or attributes, the ML technique performance is reduced to minimum percentage while the features reduced to approximately 45% of the whole attributes.

In summary, The experiment and study indicates that our methodologies might be starting point for researcher in the futures due to the promising results that are presented.

## 5.2 Future Work

The future and suggested works for the classification of breast cancer in this thesis are as follows:

1. Application of the algorithm to predict more than one type of cancer and not breast cancer in particular.
2. Work on training and examination on the database large and large based on medical images grey and colorful.
3. Apply classification with various deep learning algorithms to investigate the enhancement of classification accuracy.
4. Extract it as a front-end application (phone system or web page) for easy use in the medical field of users (doctors or patients).

## References

Aggarwal, M. (2013). Performance analysis of different feature selection methods in intrusion detection. *International Journal of Scientific & Technology Research*, *2*(6), 225–231. Retrieved from www.ijstr.org

Brownlee, J. (2018). What is the Difference Between a Batch and an Epoch in a Neural Network? Retrieved 1 April 2019, from https://machinelearningmastery.com/difference-between-a-batch-and-an-epoch/

Brownlee, J. (2019). Understand the Impact of Learning Rate on Model Performance With Deep Learning Neural Networks. Retrieved 1 April 2019, from https://machinelearningmastery.com/understand-the-dynamics-of-learning-rate-on-deep-learning-neural-networks/

Choi, J. Y. (2015). *A Generalized Multiple Classifier System for Improving Computer-aided Classification of Breast Masses in Mammography*. 251–262. https://doi.org/10.1007/s13534-015-0191-1

Deepanshu Bhalla. (n.d.). K Nearest Neighbor : Step by Step Tutorial. Retrieved 25 March 2019, from listendata website: https://www.listendata.com/2017/12/k-nearest-neighbor-step-by-step-tutorial.html

Dey, N., Borah, S., Babo, R., & Ashour, A. S. (2018). *Social Network Analytics: Computational Research Methods and Techniques*. Academic Press.

Diz, J., Marreiros, G., & Freitas, A. (2016). Applying Data Mining Techniques to Improve Breast Cancer Diagnosis. *Journal of Medical Systems*. https://doi.org/10.1007/s10916-016-0561-y

Duijm, L. E. M., Groenewoud, J. H., Jansen, F. H., Fracheboud, J., van Beek, M., & de Koning, H. J. (2004). Mammography screening in the Netherlands: delay in the diagnosis of breast cancer after breast cancer screening. *British Journal of Cancer, 91*(10), 1795.

Garg, R. (2018). *7 Types of Classification Algorithms*.

Hope, C. (2017). Seed. Retrieved 1 April 2019, from Computer Hope website: https://www.computerhope.com/jargon/s/seed.htm

Hossin, M., & Sulaiman, M. (2015). a REview on EValuation MEtrics For DAta CLassification EValuations. *International Journal of Data Mining & Knowledge Management Process (IJDKP)*, *5*(2), 1–11.

Isa, I. S., Saad, Z., Omar, S., Osman, M. K., Ahmad, K. A., & Sakim, H. A. M. (2010). Suitable MLP network activation functions for breast cancer and thyroid disease detection. *Proceedings - 2nd International Conference on Computational Intelligence, Modelling and Simulation, CIMSim 2010*, 39–44. https://doi.org/10.1109/CIMSiM.2010.93

Karabatak, M. (2015). A new classifier for breast cancer detection based on Naïve Bayesian. *Measurement: Journal of the International Measurement Confederation*, *72*, 32–36. https://doi.org/10.1016/j.measurement.2015.04.028

Khalid, S., Khalil, T., & Nasreen, S. (2014). A survey of feature selection and feature extraction techniques in machine learning. *Proceedings of 2014 Science and Information Conference, SAI 2014*, 372–378. https://doi.org/10.1109/SAI.2014.6918213

Khandelwal, R. (2019). Overview of different Optimizers for neural networks. Retrieved 1 March 2019, from https://medium.com/datadriveninvestor/overview-of-different-optimizers-for-neural-networks-e0ed119440c3

Li, J., Gao, M., & D'Agostino, R. (2019). Evaluating classification accuracy for modern learning approaches. *Statistics in Medicine*, *38*(13), 2477–2503. https://doi.org/10.1002/sim.8103

Mahajan, A., & Ganpati, A. (2014). Performance evaluation of rule based classification algorithms. *International Journal of Advanced Research in Computer Engineering & Technology*, *3*(10), 3546–3550. Retrieved from http://ijarcet.org/wp-content/uploads/IJARCET-VOL-3-ISSUE-10-3546-3550.pdf

Mohammed, M., Khan, M. B., & Bashie, E. B. M. (2016). Machine learning: Algorithms and applications. In *Machine Learning: Algorithms and Applications*. https://doi.org/10.1201/9781315371658

MOHAN, C., & NAGARAJAN, S. (2019). An improved tree model based on ensemble feature selection for classification. *TURKISH JOURNAL OF ELECTRICAL ENGINEERING & COMPUTER SCIENCES*, 1290–1307. https://doi.org/10.3906/elk-1808-85

NickGillian. (2014). *GRT: MLP*. Retrieved from http://www.nickgillian.com/wiki/pmwiki.php/GRT/MLP

Nilashi, M., Ibrahim, O., Ahmadi, H., & Shahmoradi, L. (2017). A knowledge-based system for breast cancer classification using fuzzy logic method. *Telematics and Informatics*, *34*(4), 133–144. https://doi.org/10.1016/j.tele.2017.01.007

Pendharkar, P., Rodger, J. A., Yaverbaum, G. J., Herman, N., & Benner, M. (1999). Association, statistical, mathematical and neural approaches for mining breast cancer patterns. *Expert Systems with Applications*, *17*(3), 223–232. https://doi.org/10.1016/s0957-4174(99)00036-6

Polamuri, S. (2017). How the random forest algorithm works in machine learning. Retrieved 25 March 2019, from dataaspirant website: http://dataaspirant.com/2017/05/22/random-forest-algorithm-machine-learing/

Sahu, P., & Miri, R. (2017). *A Hybrid Technique for creating classification model using Random Committee and Voted Perceptron Classifier*. *5*(Vi), 82–84.

Sayad, S. (2011). *Real Time Data Mining (Google eBook)*. Retrieved from http://books.google.com/books?id=K0WAE9xqIWAC&pgis=1

Schmidhuber, J. (2015). Deep Learning in neural networks: An overview. *Neural Networks*, *61*, 85–117. https://doi.org/10.1016/j.neunet.2014.09.003

scikit-learn.org. (n.d.). Neural network models (supervised). Retrieved 25 March 2019, from https://scikit-learn.org/stable/modules/neural_networks_supervised.html#complexity

Vafaie, H., & Imam, I. F. (1997). Feature Selection Methods : Genetic Algorithms vs . Greedy-like Search. *Proceedings of the International Conference on Fuzzy and Intelligent Control Systems*, *1*(Vafaie 93), 1–10.

Waugh, S. A., Purdie, C. A., Jordan, L. B., Vinnicombe, S., Lerski, R. A., Martin, P., & Thompson, A. M. (2015). *Magnetic resonance imaging texture analysis classification of primary breast cancer*. https://doi.org/10.1007/s00330-015-3845-6

Wolberg, WIlliam H. Mangasarian, O. (2016). *Breast Cancer Wisconsin (Diagnostic) Data Set*. Retrieved from https://www.kaggle.com/uciml/breast-cancer-wisconsin-data

Yue, W., Wang, Z., Chen, H., Payne, A., & Liu, X. (2018). Machine Learning with Applications in Breast Cancer Diagnosis and Prognosis. *Designs*, *2*(2), 13. https://doi.org/10.3390/designs2020013

Zare-Zardini, H., Amiri, A., Shanbedi, M., Taheri-Kafrani, A., Sadri, Z., Ghanizadeh, F., … Shahriari, S. (2015). Nanotechnology and Pediatric Cancer: Prevention, Diagnosis and Treatment. *Iranian Journal of Pediatric Hematology and Oncology*, *5*(4), 233–248. Retrieved from https://www.ncbi.nlm.nih.gov/pubmed/26985357

Zheng, Y., Yu, J., & Kambhamettu, C. (2007). De-enhancing the dynamic contrast-enhanced breast MRI for robust registration. *... Image Computing and ...*, *10*(Pt 1), 933–941. https://doi.org/10.1111/j.1600-6143.2008.02497.x.Plasma

# Appendix A: Result of machine learning algorithms classification for breast cancer

| families | Scheme: | Correctly Classified Instances | | Incorrectly Classified Instances | | Kappa statistic | Mean absolute error | Root mean squared error | Relative absolute error | Root relative squared error |
|---|---|---|---|---|---|---|---|---|---|---|
| bayes | Bayes Net | 546 | 95.9578% | 23 | 4.04% | 0.9136 | 0.0405 | 0.1931 | 8.67% | 39.92% |
| | Naïve Bayes | 534 | 93.8489% | 35 | 6.15% | 0.8673 | 0.0637 | 0.2452 | 13.61% | 50.72% |
| | Naïve Bayes Updateable | 534 | 93.8489% | 35 | 6.15% | 0.8673 | 0.0637 | 0.2452 | 13.61% | 50.72% |
| | Naïve Bayes Multinomial | 510 | 89.6309% | 59 | 10.36% | 0.7696 | 0.1037 | 0.3150 | 22.17% | 65.15% |
| | Naïve Bayes Multinomial Updateable | 510 | 89.6309% | 59 | 10.36% | 0.7696 | 0.1037 | 0.3150 | 22.17% | 65.15% |
| functions | Logistic | 569 | 100% | 0 | 0.00% | 1.0000 | 0.0000 | 0.0003 | 0.01% | 0.05% |
| | Simple Logistic | 560 | 98.4183% | 9 | 1.58% | **0.966** | 0.04 | 0.1232 | 8.56% | 25.48% |
| | SMO | 559 | 98.24% | 10 | 1.7575 % | 0.9621 | 0.0176 | 0.1326 | 3.76% | 27.42% |
| | SGD | 559 | 98.2425% | 10 | 1.7575 % | 0.9622 | 0.0176 | 0.1326 | 3.76% | 27.42% |
| | Multilayer Perceptron | 558 | 98.0668% | 11 | 1.93% | **0.9588** | 0.0219 | 0.1361 | 4.68% | 28.15% |
| | Voted Perceptron | 515 | 90.51% | 54 | 9.49% | 0.7978 | 0.0949 | 0.3081 | 20.29% | 63.72% |
| lazy | KNN | 569 | 100% | 0 | 0.00% | 1.0000 | 0.0018 | 0.0018 | 0.37% | 0.36% |
| | KStar | 569 | 100% | 0 | 0.00% | 1.0000 | 0.0000 % | 0.0000 | 0.00% | 0.00% |
| | LWL | 553 | 97.1880% | 16 | 2.81% | 0.9394 | 0.0753 | 0.1666 | 16.11% | 34.46% |
| meta | Multi Class Classifier | 569 | 100% | 0 | 0.00% | 1.0000 | 0.0000 | 0.0003 | 0.01% | 0.05% |
| | Random Committee | 569 | 100% | 0 | 0.00% | 1.0000 | 0.0000 | 0.0000 | 0.00% | 0.00% |

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Randomizable Filtered Classifier | 569 | 100% | 0 | 0.00% | 1.0000 | 0.0018 | 0.0018 | 0.37% | 0.36% |
| | Attribute Selected Classifier | 563 | 98.9455% | 6 | 1.0545% | 0.9774 | 0.0188 | 0.0971 | 4.03% | 20.07% |
| | Iterative Classifier Optimizer | 562 | 98.7698% | 7 | 1.2302% | 0.9736 | 0.0427 | 0.1168 | 9.14% | 24.14% |
| | LogitBoost | 562 | 98.7698% | 7 | 1.2302% | 0.9736 | 0.0427 | 0.1168 | 9.14% | 24.15% |
| | Random SubSpace | 561 | 98.59% | 8 | 1.41% | 0.9698 | 0.0534 | 0.1264 | 11.41% | 26.15% |
| | Classification Via Regression | 560 | 98.4183% | 9 | 1.5817% | 0.9659 | 0.0505 | 0.1156 | 10.80% | 23.91% |
| | Bagging | 560 | 98.4183% | 9 | 1.5817% | 0.9661 | 0.0579 | 0.1382 | 12.38% | 28.59% |
| | Filtered Classifier | 554 | 97.3638% | 15 | 2.6362% | 0.9435 | 0.0478 | 0.1546 | 10.22% | 31.97% |
| | AdaBoostM1 | 549 | 96.4851% | 20 | 3.5149% | 0.9248 | 0.0446 | 0.1503 | 9.55% | 31.09% |
| rules | PART | 565 | 99.30% | 4 | 0.70% | 0.9849 | 0.0136 | 0.0824 | 2.90% | 17.04% |
| | JRip | 561 | 98.59% | 8 | 1.41% | 0.9699 | 0.0274 | 0.1171 | 5.87% | 24.23% |
| | Decision Table | 547 | 96.13% | 22 | 3.87% | 0.9171 | 0.0814 | 0.1745 | 17.41% | 36.10% |
| trees | Random Forest | 569 | 100.00% | 0 | 0.00% | 1 | 0.0272 | 0.0655 | 5.81% | 13.55% |
| | Random Tree | 569 | 100.00% | 0 | 0.00% | 1 | 0.00% | 0 | 0.00% | 0.00% |
| | J48 | 564 | 99.12% | 5 | 0.88% | 0.9812 | 0.0165 | 0.0908 | 3.53% | 18.79% |
| | LMT | 560 | 98.42% | 9 | 1.58% | 0.966 | 0.04 | 0.1232 | 8.56% | 25.48% |

| families: | Scheme: | Precision | Recall | F-Measure | ROC Area |
|---|---|---|---|---|---|
| bayes | BayesNet | 0.9440 | 0.9480 | 0.9460 | 0.9920 |
| | NaiveBayes | 0.9360 | 0.8960 | 0.9160 | 0.9810 |
| | NaiveBayes Updateable | 0.9360 | 0.8960 | 0.9160 | 0.9810 |
| | NaiveBayes Multinomial | 0.9420 | 0.7690 | 0.8470 | 0.9440 |
| | NaiveBayes Multinomial Updateable | 0.9420 | 0.7690 | 0.8470 | 0.9440 |
| functions | Logistic | 1.0000 | 1.0000 | 1.0000 | 1.0000 |
| | SimpleLogistic | 0.9900 | 0.9670 | 0.9790 | 0.9980 |
| | SMO | 0.9950 | 0.9580 | 0.9760 | 0.9770 |
| | SGD | 0.9900 | 0.9620 | 0.9760 | 0.9780 |
| | Multilayer Perceptron | 0.9670 | 0.9810 | 0.9740 | 0.9930 |
| | Voted Perceptron | 0.8660 | 0.8820 | 0.8740 | 0.9020 |
| lazy | KNN | 1.0000 | 1.0000 | 1.0000 | 1.0000 |
| | KStar | 1.0000 | 1.0000 | 1.0000 | 1.0000 |
| | LWL | 0.9800 | 0.9430 | 0.9620 | 0.9810 |
| meta | MultiClassClassifier | 1.0000 | 1.0000 | 1.0000 | 1.0000 |
| | Random Committee | 1.0000 | 1.0000 | 1.0000 | 1.0000 |
| | Randomizable Filtered Classifier | 1.0000 | 1.0000 | 1.0000 | 1.0000 |
| | Attribute Selected Classifier | 0.9900 | 0.9810 | 0.9860 | 0.9930 |
| | Iterative Classifier Optimizer | 0.9900 | 0.9760 | 0.9830 | 0.9980 |
| | LogitBoost | 0.9900 | 0.9760 | 0.9830 | 0.9980 |
| | RandomSubSpace | 0.9950 | 0.9670 | 0.9810 | 0.9930 |

| | | | | | |
|---|---|---|---|---|---|
| | Classification Via Regression | 0.9950 | 0.9620 | 0.9780 | 1.0000 |
| | Bagging | 0.9860 | 0.9720 | 0.9790 | 0.9960 |
| | FilteredClassifier | 0.9710 | 0.9580 | 0.9640 | 0.9820 |
| | AdaBoostM1 | 0.9530 | 0.9530 | 0.9530 | 0.9960 |
| rules | PART | 0.9950 | 0.9860 | 0.9910 | 0.9950 |
| | JRip | 0.9860 | 0.9760 | 0.9810 | 0.9860 |
| | DecisionTable | 0.9520 | 0.9430 | 0.9480 | 0.9890 |
| trees | RandomForest | 1.0000 | 1.0000 | 1.0000 | 1.0000 |
| | RandomTree | 1.0000 | 1.0000 | 1.0000 | 1.0000 |
| | J48 | 0.9950 | 0.9810 | 0.9880 | 0.9930 |
| | LMT | 0.9900 | 0.9670 | 0.9790 | 0.9980 |

# Appendix B: some screenshots from WDBC dataset

| id | radius_mean | texture_mean | perimeter_mean | area_mean | smoothness_mean | compactness_mean | concavity_mean |
|---|---|---|---|---|---|---|---|
| **1** | 17.99 | 10.38 | 122.8 | 1001 | 0.1184 | 0.2776 | 0.3001 |
| **2** | 20.57 | 17.77 | 132.9 | 1326 | 0.08474 | 0.07864 | 0.0869 |
| **3** | 19.69 | 21.25 | 130 | 1203 | 0.1096 | 0.1599 | 0.1974 |
| **4** | 11.42 | 20.38 | 77.58 | 386.1 | 0.1425 | 0.2839 | 0.2414 |
| **5** | 20.29 | 14.34 | 135.1 | 1297 | 0.1003 | 0.1328 | 0.198 |
| **6** | 12.45 | 15.7 | 82.57 | 477.1 | 0.1278 | 0.17 | 0.1578 |
| **7** | 18.25 | 19.98 | 119.6 | 1040 | 0.09463 | 0.109 | 0.1127 |
| **8** | 13.71 | 20.83 | 90.2 | 577.9 | 0.1189 | 0.1645 | 0.09366 |
| **9** | 13 | 21.82 | 87.5 | 519.8 | 0.1273 | 0.1932 | 0.1859 |
| **10** | 12.46 | 24.04 | 83.97 | 475.9 | 0.1186 | 0.2396 | 0.2273 |
| **11** | 16.02 | 23.24 | 102.7 | 797.8 | 0.08206 | 0.06669 | 0.03299 |
| **12** | 15.78 | 17.89 | 103.6 | 781 | 0.0971 | 0.1292 | 0.09954 |
| **13** | 19.17 | 24.8 | 132.4 | 1123 | 0.0974 | 0.2458 | 0.2065 |
| **14** | 15.85 | 23.95 | 103.7 | 782.7 | 0.08401 | 0.1002 | 0.09938 |
| **15** | 13.73 | 22.61 | 93.6 | 578.3 | 0.1131 | 0.2293 | 0.2128 |
| **16** | 14.54 | 27.54 | 96.73 | 658.8 | 0.1139 | 0.1595 | 0.1639 |
| **17** | 14.68 | 20.13 | 94.74 | 684.5 | 0.09867 | 0.072 | 0.07395 |
| **18** | 16.13 | 20.68 | 108.1 | 798.8 | 0.117 | 0.2022 | 0.1722 |
| **19** | 19.81 | 22.15 | 130 | 1260 | 0.09831 | 0.1027 | 0.1479 |
| **20** | 13.54 | 14.36 | 87.46 | 566.3 | 0.09779 | 0.08129 | 0.06664 |
| **21** | 13.08 | 15.71 | 85.63 | 520 | 0.1075 | 0.127 | 0.04568 |
| **22** | 9.5 | 12.44 | 60.34 | 273.9 | 0.1024 | 0.06492 | 0.02956 |
| **23** | 15.34 | 14.26 | 102.5 | 704.4 | 0.1073 | 0.2135 | 0.2077 |
| **24** | 21.16 | 23.04 | 137.2 | 1404 | 0.09428 | 0.1022 | 0.1097 |

| smoothness_se | compactness_se | concavity_se | concave points_se | symmetry_se | fractal_dimension_se | radius_worst |
|---|---|---|---|---|---|---|
| **0.006399** | 0.04904 | 0.05373 | 0.01587 | 0.03003 | 0.006193 | 25.38 |
| **0.005225** | 0.01308 | 0.0186 | 0.0134 | 0.01389 | 0.003532 | 24.99 |
| **0.00615** | 0.04006 | 0.03832 | 0.02058 | 0.0225 | 0.004571 | 23.57 |
| **0.00911** | 0.07458 | 0.05661 | 0.01867 | 0.05963 | 0.009208 | 14.91 |
| **0.01149** | 0.02461 | 0.05688 | 0.01885 | 0.01756 | 0.005115 | 22.54 |
| **0.00751** | 0.03345 | 0.03672 | 0.01137 | 0.02165 | 0.005082 | 15.47 |
| **0.004314** | 0.01382 | 0.02254 | 0.01039 | 0.01369 | 0.002179 | 22.88 |
| **0.008805** | 0.03029 | 0.02488 | 0.01448 | 0.01486 | 0.005412 | 17.06 |
| **0.005731** | 0.03502 | 0.03553 | 0.01226 | 0.02143 | 0.003749 | 15.49 |
| **0.007149** | 0.07217 | 0.07743 | 0.01432 | 0.01789 | 0.01008 | 15.09 |
| **0.004029** | 0.009269 | 0.01101 | 0.007591 | 0.0146 | 0.003042 | 19.19 |
| **0.005771** | 0.04061 | 0.02791 | 0.01282 | 0.02008 | 0.004144 | 20.42 |
| **0.003139** | 0.08297 | 0.0889 | 0.0409 | 0.04484 | 0.01284 | 20.96 |
| **0.009769** | 0.03126 | 0.05051 | 0.01992 | 0.02981 | 0.003002 | 16.84 |
| **0.006429** | 0.05936 | 0.05501 | 0.01628 | 0.01961 | 0.008093 | 15.03 |
| **0.005607** | 0.0424 | 0.04741 | 0.0109 | 0.01857 | 0.005466 | 17.46 |
| **0.005718** | 0.01162 | 0.01998 | 0.01109 | 0.0141 | 0.002085 | 19.07 |
| **0.007026** | 0.02501 | 0.03188 | 0.01297 | 0.01689 | 0.004142 | 20.96 |
| **0.006494** | 0.01893 | 0.03391 | 0.01521 | 0.01356 | 0.001997 | 27.32 |
| **0.008462** | 0.0146 | 0.02387 | 0.01315 | 0.0198 | 0.0023 | 15.11 |
| **0.004097** | 0.01898 | 0.01698 | 0.00649 | 0.01678 | 0.002425 | 14.5 |
| **0.009606** | 0.01432 | 0.01985 | 0.01421 | 0.02027 | 0.002968 | 10.23 |
| **0.006789** | 0.05328 | 0.06446 | 0.02252 | 0.03672 | 0.004394 | 18.07 |
| **0.004728** | 0.01259 | 0.01715 | 0.01038 | 0.01083 | 0.001987 | 29.17 |

| **texture_worst** | **perimeter_worst** | **area_worst** | **smoothness_worst** | **compactness_worst** | **concavity_worst** |
|---|---|---|---|---|---|
| **17.33** | 184.6 | 2019 | 0.1622 | 0.6656 | 0.7119 |
| **23.41** | 158.8 | 1956 | 0.1238 | 0.1866 | 0.2416 |
| **25.53** | 152.5 | 1709 | 0.1444 | 0.4245 | 0.4504 |
| **26.5** | 98.87 | 567.7 | 0.2098 | 0.8663 | 0.6869 |
| **16.67** | 152.2 | 1575 | 0.1374 | 0.205 | 0.4 |
| **23.75** | 103.4 | 741.6 | 0.1791 | 0.5249 | 0.5355 |
| **27.66** | 153.2 | 1606 | 0.1442 | 0.2576 | 0.3784 |
| **28.14** | 110.6 | 897 | 0.1654 | 0.3682 | 0.2678 |
| **30.73** | 106.2 | 739.3 | 0.1703 | 0.5401 | 0.539 |
| **40.68** | 97.65 | 711.4 | 0.1853 | 1.058 | 1.105 |
| **33.88** | 123.8 | 1150 | 0.1181 | 0.1551 | 0.1459 |
| **27.28** | 136.5 | 1299 | 0.1396 | 0.5609 | 0.3965 |
| **29.94** | 151.7 | 1332 | 0.1037 | 0.3903 | 0.3639 |
| **27.66** | 112 | 876.5 | 0.1131 | 0.1924 | 0.2322 |
| **32.01** | 108.8 | 697.7 | 0.1651 | 0.7725 | 0.6943 |
| **37.13** | 124.1 | 943.2 | 0.1678 | 0.6577 | 0.7026 |
| **30.88** | 123.4 | 1138 | 0.1464 | 0.1871 | 0.2914 |
| **31.48** | 136.8 | 1315 | 0.1789 | 0.4233 | 0.4784 |
| **30.88** | 186.8 | 2398 | 0.1512 | 0.315 | 0.5372 |
| **19.26** | 99.7 | 711.2 | 0.144 | 0.1773 | 0.239 |
| **20.49** | 96.09 | 630.5 | 0.1312 | 0.2776 | 0.189 |
| **15.66** | 65.13 | 314.9 | 0.1324 | 0.1148 | 0.08867 |
| **19.08** | 125.1 | 980.9 | 0.139 | 0.5954 | 0.6305 |
| **35.59** | 188 | 2615 | 0.1401 | 0.26 | 0.3155 |

| concave points_mean | symmetry_mean | fractal_dimension_mean | radius_se | texture_se | perimeter_se | area_se |
|---|---|---|---|---|---|---|
| **0.1471** | 0.2419 | 0.07871 | 1.095 | 0.9053 | 8.589 | 153.4 |
| **0.07017** | 0.1812 | 0.05667 | 0.5435 | 0.7339 | 3.398 | 74.08 |
| **0.1279** | 0.2069 | 0.05999 | 0.7456 | 0.7869 | 4.585 | 94.03 |
| **0.1052** | 0.2597 | 0.09744 | 0.4956 | 1.156 | 3.445 | 27.23 |
| **0.1043** | 0.1809 | 0.05883 | 0.7572 | 0.7813 | 5.438 | 94.44 |
| **0.08089** | 0.2087 | 0.07613 | 0.3345 | 0.8902 | 2.217 | 27.19 |
| **0.074** | 0.1794 | 0.05742 | 0.4467 | 0.7732 | 3.18 | 53.91 |
| **0.05985** | 0.2196 | 0.07451 | 0.5835 | 1.377 | 3.856 | 50.96 |
| **0.09353** | 0.235 | 0.07389 | 0.3063 | 1.002 | 2.406 | 24.32 |
| **0.08543** | 0.203 | 0.08243 | 0.2976 | 1.599 | 2.039 | 23.94 |
| **0.03323** | 0.1528 | 0.05697 | 0.3795 | 1.187 | 2.466 | 40.51 |
| **0.06606** | 0.1842 | 0.06082 | 0.5058 | 0.9849 | 3.564 | 54.16 |
| **0.1118** | 0.2397 | 0.078 | 0.9555 | 3.568 | 11.07 | 116.2 |
| **0.05364** | 0.1847 | 0.05338 | 0.4033 | 1.078 | 2.903 | 36.58 |
| **0.08025** | 0.2069 | 0.07682 | 0.2121 | 1.169 | 2.061 | 19.21 |
| **0.07364** | 0.2303 | 0.07077 | 0.37 | 1.033 | 2.879 | 32.55 |
| **0.05259** | 0.1586 | 0.05922 | 0.4727 | 1.24 | 3.195 | 45.4 |
| **0.1028** | 0.2164 | 0.07356 | 0.5692 | 1.073 | 3.854 | 54.18 |
| **0.09498** | 0.1582 | 0.05395 | 0.7582 | 1.017 | 5.865 | 112.4 |
| **0.04781** | 0.1885 | 0.05766 | 0.2699 | 0.7886 | 2.058 | 23.56 |
| **0.0311** | 0.1967 | 0.06811 | 0.1852 | 0.7477 | 1.383 | 14.67 |
| **0.02076** | 0.1815 | 0.06905 | 0.2773 | 0.9768 | 1.909 | 15.7 |
| **0.09756** | 0.2521 | 0.07032 | 0.4388 | 0.7096 | 3.384 | 44.91 |
| **0.08632** | 0.1769 | 0.05278 | 0.6917 | 1.127 | 4.303 | 93.99 |

| concave points_worst | symmetry_worst | fractal_dimension_worst | diagnosis |
|---|---|---|---|
| **0.2654** | 0.4601 | 0.1189 | M |
| **0.186** | 0.275 | 0.08902 | M |
| **0.243** | 0.3613 | 0.08758 | M |
| **0.2575** | 0.6638 | 0.173 | M |
| **0.1625** | 0.2364 | 0.07678 | M |
| **0.1741** | 0.3985 | 0.1244 | M |
| **0.1932** | 0.3063 | 0.08368 | M |
| **0.1556** | 0.3196 | 0.1151 | M |
| **0.206** | 0.4378 | 0.1072 | M |
| **0.221** | 0.4366 | 0.2075 | M |
| **0.09975** | 0.2948 | 0.08452 | M |
| **0.181** | 0.3792 | 0.1048 | M |
| **0.1767** | 0.3176 | 0.1023 | M |
| **0.1119** | 0.2809 | 0.06287 | M |
| **0.2208** | 0.3596 | 0.1431 | M |
| **0.1712** | 0.4218 | 0.1341 | M |
| **0.1609** | 0.3029 | 0.08216 | M |
| **0.2073** | 0.3706 | 0.1142 | M |
| **0.2388** | 0.2768 | 0.07615 | M |
| **0.1288** | 0.2977 | 0.07259 | B |
| **0.07283** | 0.3184 | 0.08183 | B |
| **0.06227** | 0.245 | 0.07773 | B |
| **0.2393** | 0.4667 | 0.09946 | M |
| **0.2009** | 0.2822 | 0.07526 | M |

Where is  M: Malignant, and B: Benign.

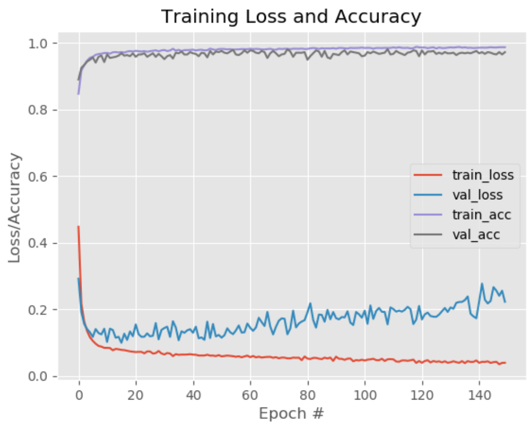
# Appendix C: Figures of Experiments Results

## One Hidden Layer

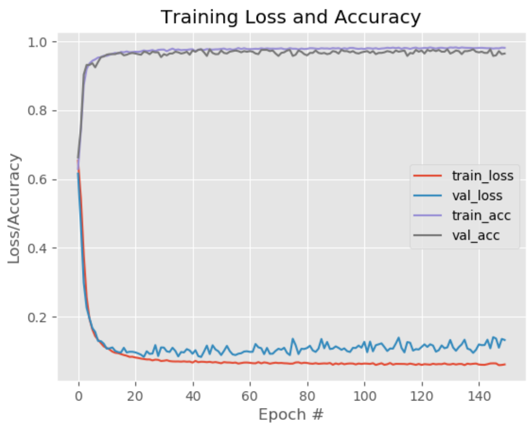

Optimizer ‘Adam’, activation function1 ‘ relu’
acc= 0.9613


Optimizer ‘Adam’, activation function1 ‘sigmoid’
acc= 0.97


Optimizer ‘Adam’,activation function1 ‘tanh’
acc=0.9665




Optimizer ‘Rmsprop’,activation function1 ‘relu’
acc= 0.9735



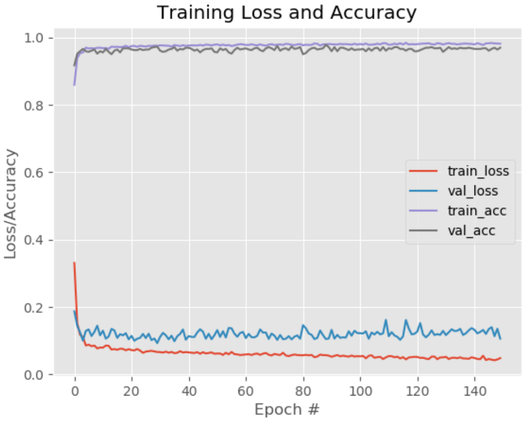

Optimizer ‘Rmsprop’, activation function1 ‘sigmoid’
acc= 0.9647


Optimizer ‘Rmsprop’,activation function1 ‘tanh’
acc= 0.9665

# Two hidden layers


Optimizer ‘Adam’,activation function1 ‘ relu’
acc= 0.9683

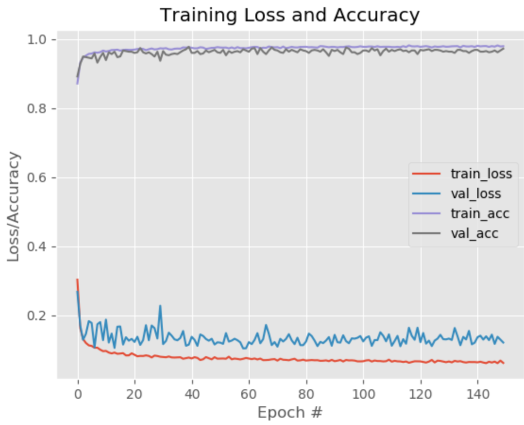

Optimizer ‘Adam’,activation function1 ‘sigmoid’
acc= 0.97




Optimizer ‘Adam’,activation function1 ‘tanh’
acc= 0.97

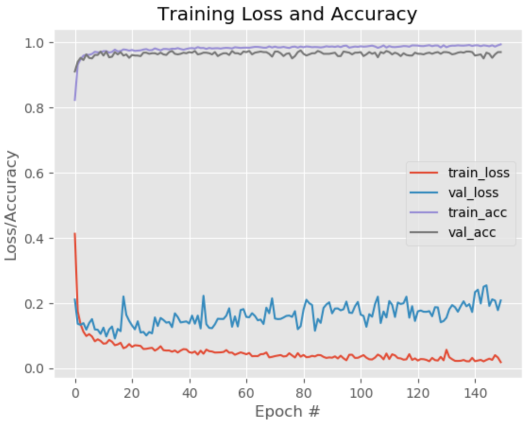

Optimizer ‘Rmsprop’,activation function1 ‘relu’
acc= 0.9753


Optimizer ‘Rmsprop’,activation function1 ‘sigmoid’
acc= 0.9718

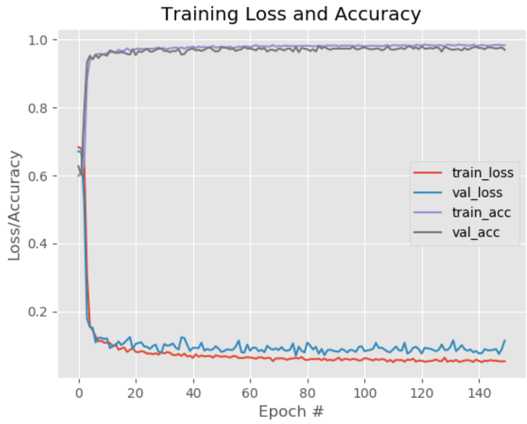



Optimizer ‘Rmsprop’,activation function1 ‘tanh’
acc= 0.9718

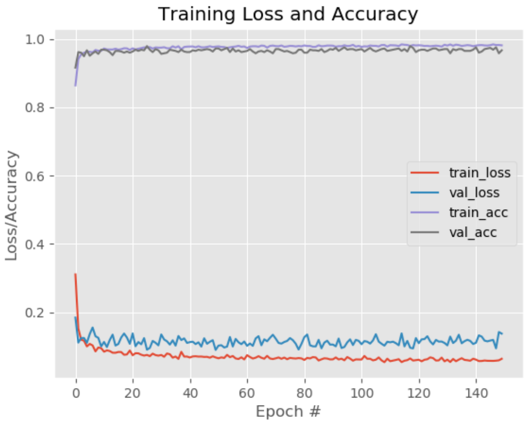
# Four hidden layers



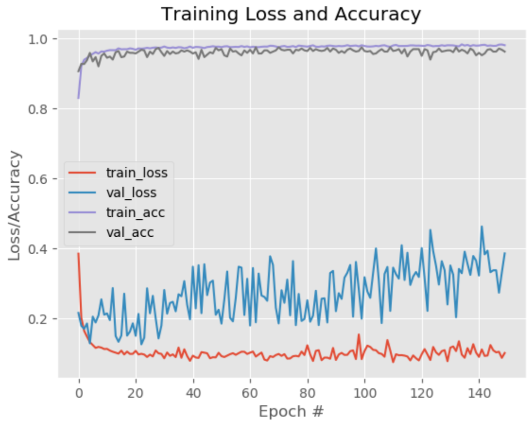

Optimizer 'Adam', activation function1 ' relu' acc= 0.9701




Optimizer 'Adam', activation function1 'sigmoid' acc= 0.9699

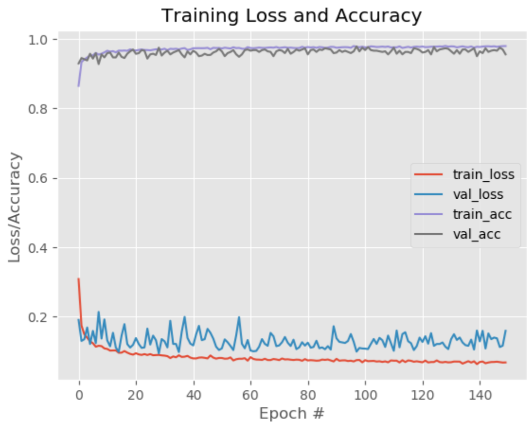



Optimizer 'Adam', activation function1 'tanh' acc= 0.9665



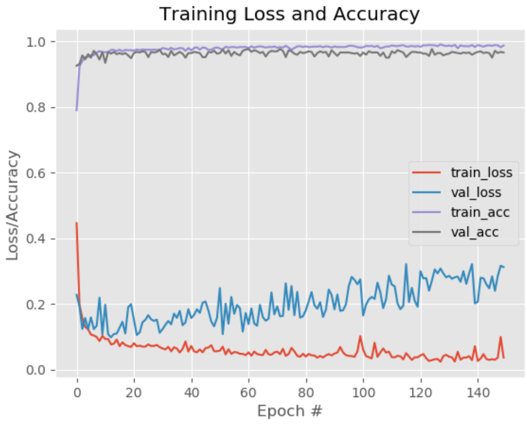

Optimizer 'Rmsprop',activation function1 'relu' acc= 0.963


Optimizer 'Rmsprop', activation function1 'sigmoid' acc= 0.9752

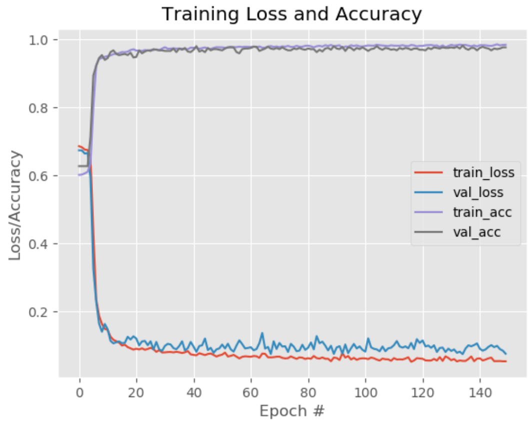



Optimizer 'Rmsprop',activation function1 'tanh' acc= 0.9561

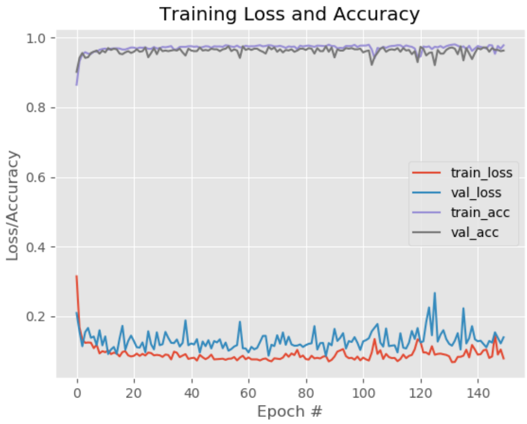
# Six hidden layers



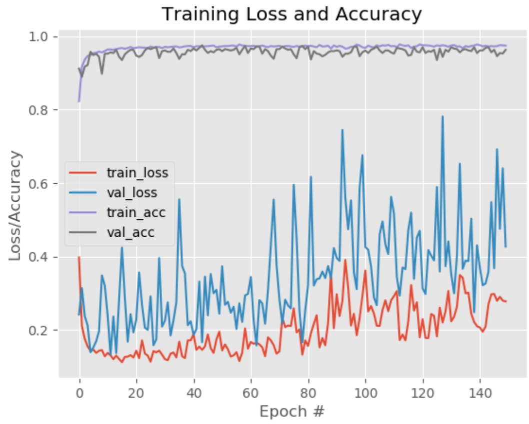

Optimizer ‘Adam’, activation function1 ‘ relu’
acc= 0.9665



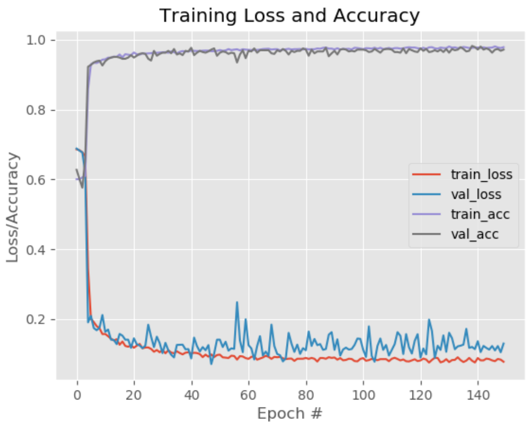

Optimizer ‘Adam’, activation function1 ‘sigmoid’
acc= 0.9699



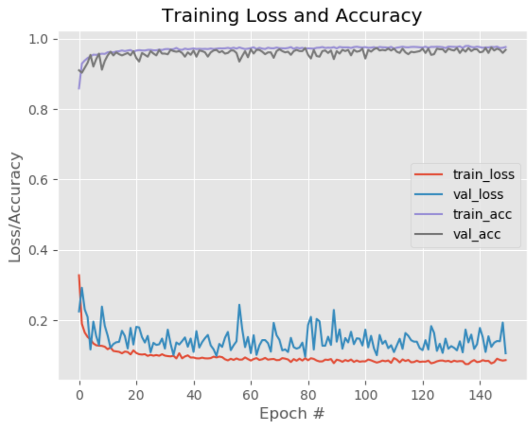

Optimizer ‘Adam’, activation function1 ‘tanh’
acc= 0.963




Optimizer ‘Rmsprop’,activation function1 ‘relu’
acc= 0.963

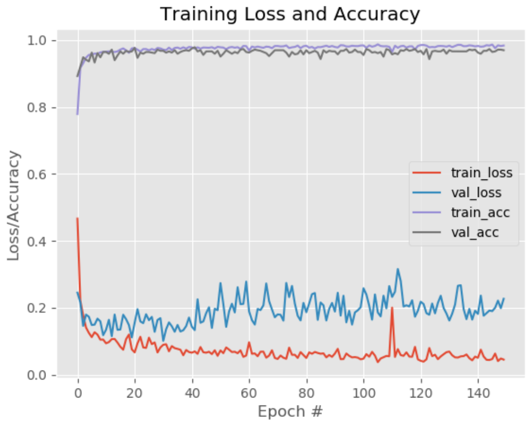



Optimizer ‘Rmsprop’,activation function1 ‘sigmoid’
acc= 0.9718




Optimizer ‘Rmsprop’,activation function1 ‘tanh’
acc= 0.9683

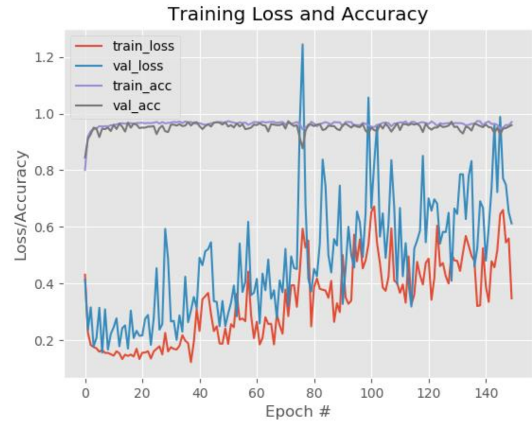
## Eight hidden layers



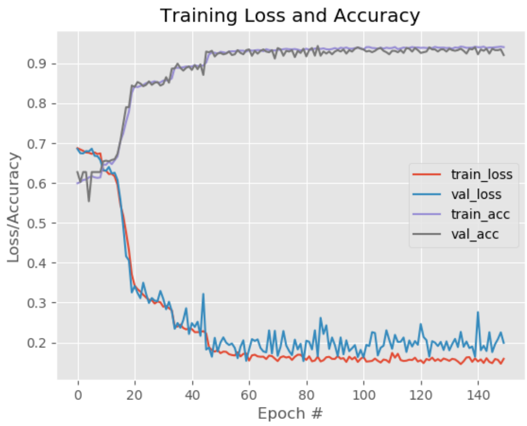

Optimizer ‘Adam’, activation function1 ‘ relu’
acc=0.970


Optimizer ‘Adam’, activation function1 ‘sigmoid’
acc=0.695


Optimizer ‘Adam’, activation function1 ‘tanh’
acc= 0.656




Optimizer ‘Rmsprop’,activation function1 ‘relu’
acc= 0.959

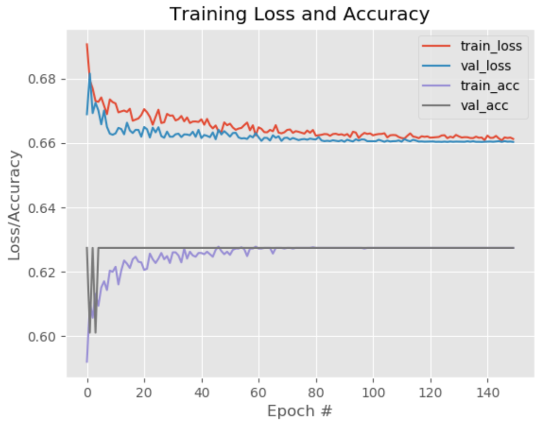



Optimizer ‘Rmsprop’,activation function1 ‘sigmoid’
acc= 0.921

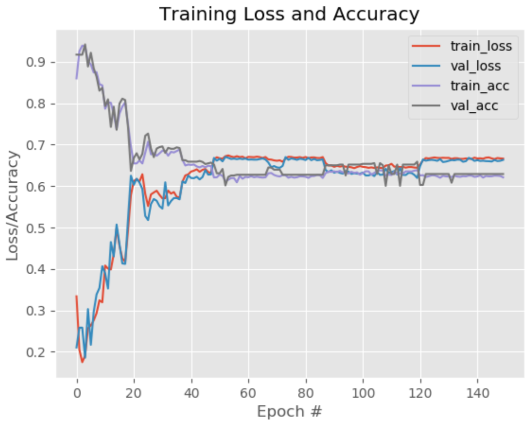

Optimizer ‘Rmsprop’,activation function1 ‘tanh’
acc= 0.940

## Ten hidden layers

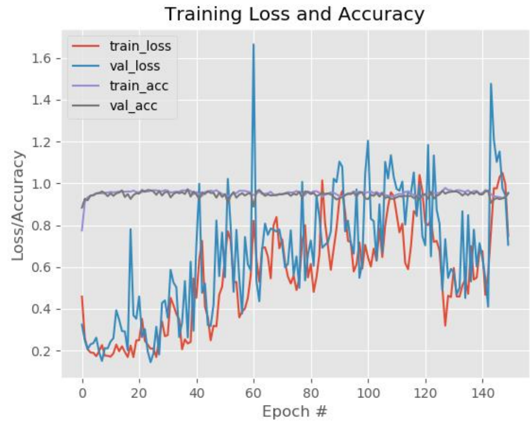

Optimizer ‘Adam’, activation function1 ‘ relu’
acc= 0.961



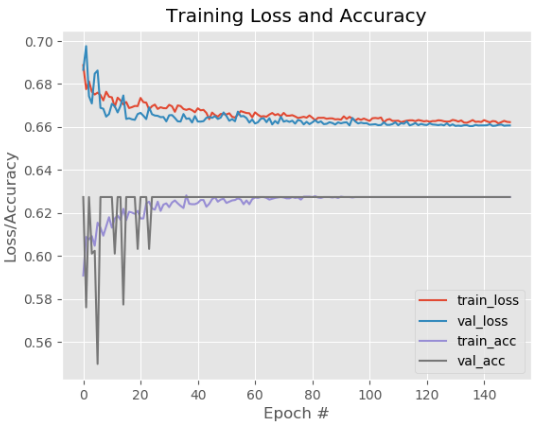

Optimizer ‘Adam’, activation function1 ‘sigmoid’
acc= 0.627



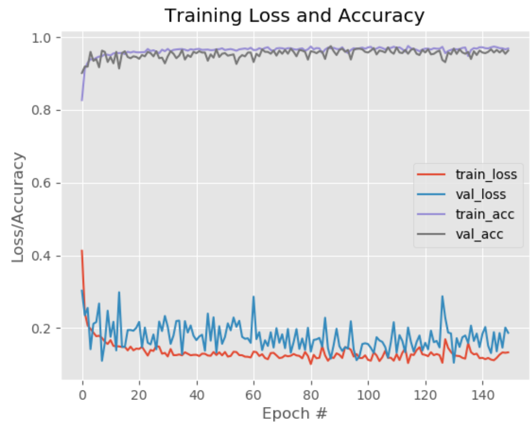

Optimizer ‘Adam’, activation function1 ‘tanh’
acc= 0.629




Optimizer ‘Rmsprop’,activation function1 ‘relu’
acc= 0.954




Optimizer ‘Rmsprop’,activation function1 ‘sigmoid’
acc= 0.627




Optimizer ‘Rmsprop’,activation function1 ‘tanh’
acc= 0.963